

클라우드 시스템 및 인증체제 구현

팀 명 : 클라우드 캐슬
지도 교수 : 양환석 교수님
팀 장 : 강지호
팀 원 : 최진석
양유리
오수빈

2019. 10.

중부대학교 정보보호학과

목 차

1. 서론

1.1 연구 배경 및 주제 선정	2
1.2 연구 목적	2

2. 관련 연구

2.1 CentOS	2
2.2 Linux	3
2.2 Apache	3
2.3 TomCat	3
2.4 Oracle Database	3
2.5 JSP	3
2.6 Hadoop	4
2.7 Kerberos	4

3. 본론

3.1 시스템 구성	4
3.2 시스템 구상도	6
3.2.1 로그인과 업로드 및 다운로드 구상도	6
3.2.2 인증시스템 구상도	6
3.3 프로그램 구성	8
3.3.1 웹페이지 기능	8
3.3.2 인증시스템 기능	9

4. 결론

.....	10
-------	----

5. 별첨

5.1 발표 PPT	11
5.2 소스코드	25

1. 서론

1.1 연구 배경 및 주제 선정

클라우드 는 요즘 시대에는 없어서는 안 되는 필수 파일 및 문서들의 데이터 저장소이다. 최근 개별 회사에서는 서버를 두는 대신 대용량 데이터 센터의 저장 공간을 빌려 쓰는 서비스를 사용하며 이것으로 제일 큰 장점은 인터넷이 되는 모든 곳에서 저장하고 다운로드를 받을 수 있어 어디서든 업무가 가능한 것이다. 그렇지만 이렇게 어디서든 파일을 다운로드 받고 볼 수 있다는 장점 때문에 기업들의 중요한 기밀들이 담긴 것들이 위험에 노출되어있다. 이런 클라우드 데이터 센터는 범외자들에게 매우 매력적인 시장이 되어 버린다. 그렇기에 회사내에서는 어디서든 쓸 수 있고 밖으로 새가지 않도록 하는 보안을 강화한 클라우드 서비스를 만들어보려고 생각하게 되었다.

1.2 연구 목적

우리의 클라우드인 JB-HARD는 회사의 직급에 따라 보여 지는 문서자체를 제한시키며 문서들이 저장 될 때에는 하둡 서비스를 사용해 데이터노드에 다 분산되어 저장되어지게 한다. 그리고 사용자들이 허가된 사용자인지 어떤 사용자인지 확인 할 수 있도록 하는 로그인 과정이 두 가지로 나뉘어진다. 첫 번째로는 허가된 사용자인지 확인할 수 있는 인증 로그인 과정이다. 관리자에게 발급받은 ID와 비밀번호로 로그인을 하며 두 번째로 직급 로그인 이다. 이 또한 관리자에게 발급받은 ID와 비밀번호이며 이 때 ID는 그 전 로그인 과정에서 ID는 같지만 비밀번호는 다르다. 로그인한 사용자가 무슨 직급을 가졌으면 어떤 서비스를 사용할 수 있는 지에 대한 직급권한 로그인 과정을 거치게 되는 시나리오를 구상으로 사용자에 대한 권한을 강화하는 개인 회사의 클라우드 시스템을 구상하였다.

2. 관련 연구

2.1 CentOS

Centos(Community Enterprise Operating System)는 센트OS 프로젝트에서 레드햇 제휴로 개발한 컴퓨터 운영 체제이다. 업스트림 소스인 레드햇 엔터프라이즈 리눅스와 완벽하게 호환되는 무료 기업용 컴퓨팅 플랫폼을 제공할 목적으로 만들어진 리눅스계 운영 체제 가운데 하나다 특징으로는 무료로 사용 가능해서 서버용으로 리눅스를 운영하기에 매우 좋다.

2.2 Linux

리눅스 토르발스가 커뮤니티 주체로 개발한 컴퓨터 운영 체제이다. 혹은 커널을 뜻하기도 한다. 리눅스는 자유 소프트웨어와 오픈 소스 개발의 가장 유명한 표본으로 들 수 있다. 리눅스는 다중 사용자, 다중 작업(멀티태스킹), 다중 스레드를 지원하는 네트워크 운영 체제(NOS)이다.

2.3 Apache

아파치(apache)란? 세계에서 가장 많이 쓰는 웹 서버중 하나이며, 아파치 소프트웨어 재단에서 관리하는 HTTP 웹 서버 이다.

Apache는 Apache재단에서 만든 HTTP서버로 워낙 다양한 추가기능에, 구축이 쉽다는 이유 때문에 많이 쓰고 있습니다 하지만 Apache 자체만으로 엄청 무겁고, Squid와 함께 Slowloris 취약점이 발견되었기에, 보통 프로그래밍 능력이 능숙한 사람들이나, 대형사이트 운영자는 Nginx, IIS를 주로 쓰고있지만 대부분의 중소기업들은 무료이기때문에 많이 쓰인다

2.4 Tomcat

현재 가장 일반적이고 많이 사용되는 WAS(web application server)이며, 동적인 웹을 만들기 위한 웹 컨테이너, 서블릿 컨테이너라고 불린다. 웹 서버와 연동하여 실행할 수 있는 자바 환경을 제공하여 자바 서버 페이지(JSP)와 자바 서블릿이 실행할 수 있는 환경을 제공하고 있습니다. 톰캣은 관리툴을 통해 설정을 변경할 수 있지만, XML 파일을 편집하여 설정할 수도 있습니다. 또한, DB연결,데이터 조작, 다른 응용프로그램과 상호 작용이 가능하고, 톰캣은 8080포트로 처리한다.

2.5 Oracle Database

미국 오라클(Oracle)사의 관계형 데이터베이스 관리 시스템의 이름이다. 현재 유닉스 환경에서 가장 널리 사용되는 RDBMS이다. 검색이나 업데이트용 언어로는 국제표준화기구의 표준 구조화 조회 언어와 PL/SQL을 지원한다.

2.6 JSP

Java 언어를 기반으로 하는 Server Side 스크립트 언어이며, HTML 코드에 Java 코드를 넣어 동적인 웹 페이지를 생성하는 웹 어플리케이션 도구 JSP를 통해 정적인 HTML과 동적으로 생성된 contents(HTTP 요청 파라미터)를 혼합하여 사용할 수 있다. 즉, 사용자가 입력한 contents에 맞게 동적인 웹 페이지를 생성한다. 특징으로는 스크립트 언어이기 때문에 자바 기능을 그대로 사용할 수 있고, 사용자 정의 태그(custom tags)를 사용하여, 보다 효율적으로 웹 사이트를 구성할 수 있고, HTML 코드 안에 Java 코드가 있기 때문에 HTML 코드를 작성하기 쉽다.

2.7 Hadoop

하둡이란 여러 개의 컴퓨터를 하나로 묶어 대용량데이터를 처리하는 기술로, 수천대의 분산된 x86장비에 대용량 파일을 저장할 수 있는 기능을 제공하는 분산파일 시스템과, 저장된 파일을 분산된 서버의 CPU와 메모리 자원을 이용하여 빠르게 분석하는 맵리듀스 플랫폼으로 시작되었으나, 여러 데이터저장, 실행엔진, 프로그래밍 및 데이터처리 같은 하둡 생태계 전반을 포함하는 의미로 확장 발전 되었다.

하둡은 대용량 데이터를 적은비용으로 더 빠르게 분석할 수 있는 소프트웨어이며, 빅 데이터 처리와 분석을 위한 플랫폼 중 사실상 표준으로 자리잡고 있다.

여러 대의 컴퓨터로 데이터를 분석하고 저장하는 방식으로 분석에 필요했던 많은 비용과 시간을 단축 할수 있게 되었다. 페이스북 북의 자동 이미지검색, 금융거래 내역 분석을 통한 사기방지, 검색 패턴을 통한 광고 타겟 및 마케팅 등 여러 분야에서 활용 될수 있다.

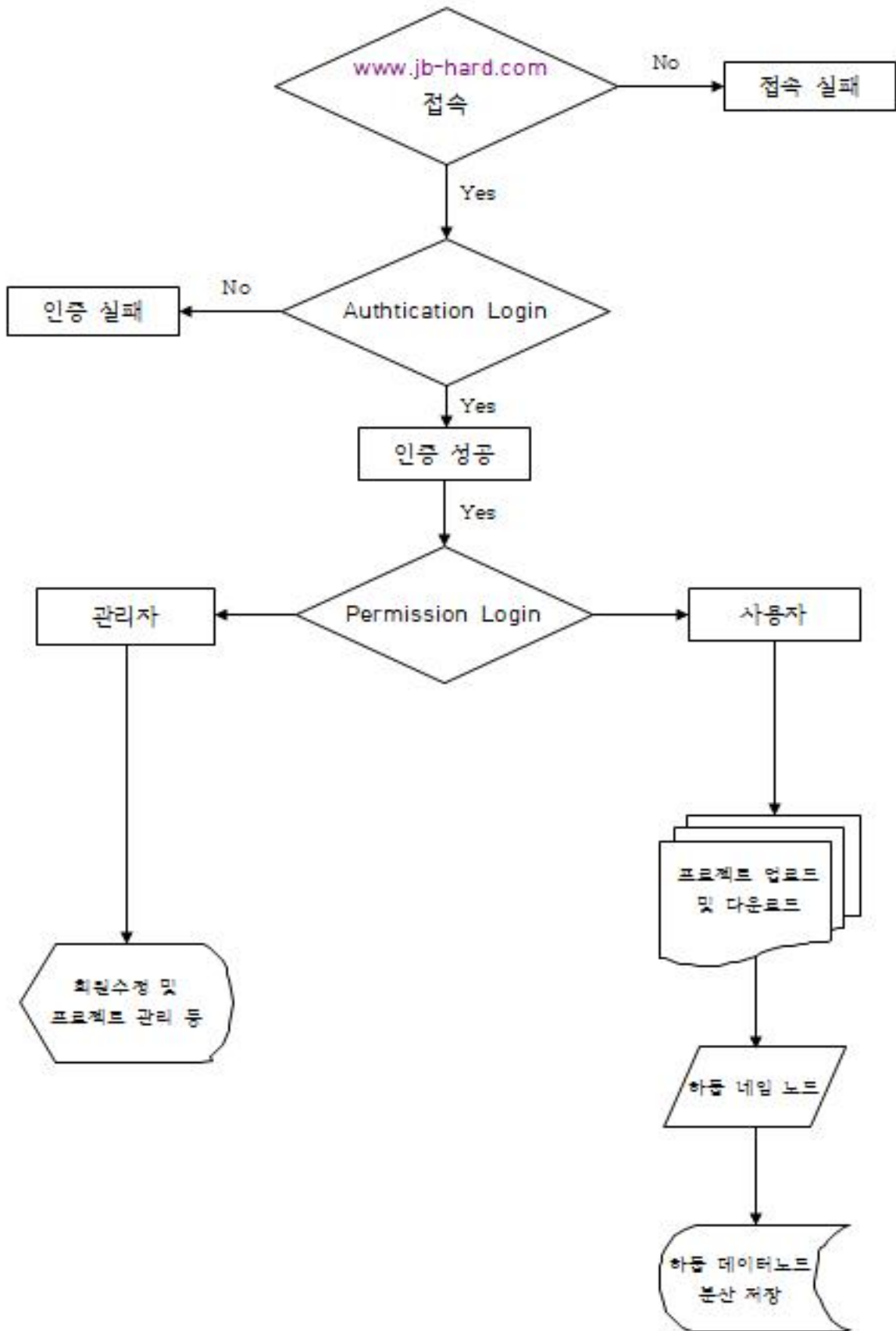
2.8 Kerberos

Kerberos는 개방 네트워크상에서 인증과 통신의 암호화를 시행하여 보안성을 확보하기 위한 알고리즘으로 그리스 신화에서 Kerberos가 저승의 입구를 지키는 머리 3개 달린 개를 말하듯이 MIT에서 Athena 프로젝트의 일환으로 개발될 당시에는 네트워크의 입구를 보호하기 위하여 인증, 계정, 감사의 3가지 요소를 갖는 것을 목적으로 했으나 인증부분만이 완성될 수 있었다. Kerberos는 신뢰할 수 있는 제 3기관인 키 분배센터(KDC)가 클라이언트의 패스워드를 기초로 생성한 티켓을(TGT: ticket-granting ticket) 발급하고 클라이언트는 접근할 서버에 사용함으로써 패스워드의 누출 위험을 줄이고 보다 높은 상호 인증을 구현하는 알고리즘이다. 다른 인증법과는 달리 공개키 암호화 알고리즘을 사용하는 것이 아니라 비밀 키 알고리즘만을 사용한다.

3. 본론

3.1 시스템 구성

웹 사이트에 로그인할 때 1차 사용자 인증, 2차 권한 인증을 하게 된다. 사용자 인증은 관리자에게 발급받은 ID와 비밀번호로 로그인하면 KDC에서 발급받은 티켓을 통한 인증 과정이 거쳐서 인증이 성공하면 티켓의 유효기간 동안은 재인증할 필요가 없게 된다. 그 다음 DB를 통한 권한 인증으로 이 사용자가 어떤 사용자인지 파악하고 권한에 따라 볼 수 있는 프로젝트가 정해지게 된다. 로그인 후에 관리자는 프로젝트 및 회원을 추가 삭제 할 수 있으며, 사용자들은 프로젝트 내에 파일을 업로드/다운로드를 할 수 있다. 이 때 저장되는 파일은 하둡 내에 네임노드서버를 거쳐 데이터 노드 서버에 분산처리 되어 저장되기 때문에 많은 양의 파일을 저장할 수 있다.



[그림 1. 프로그램 동작 과정]

3.2 시스템 구상도

3.2.1 로그인과 업로드 및 다운로드 구상도

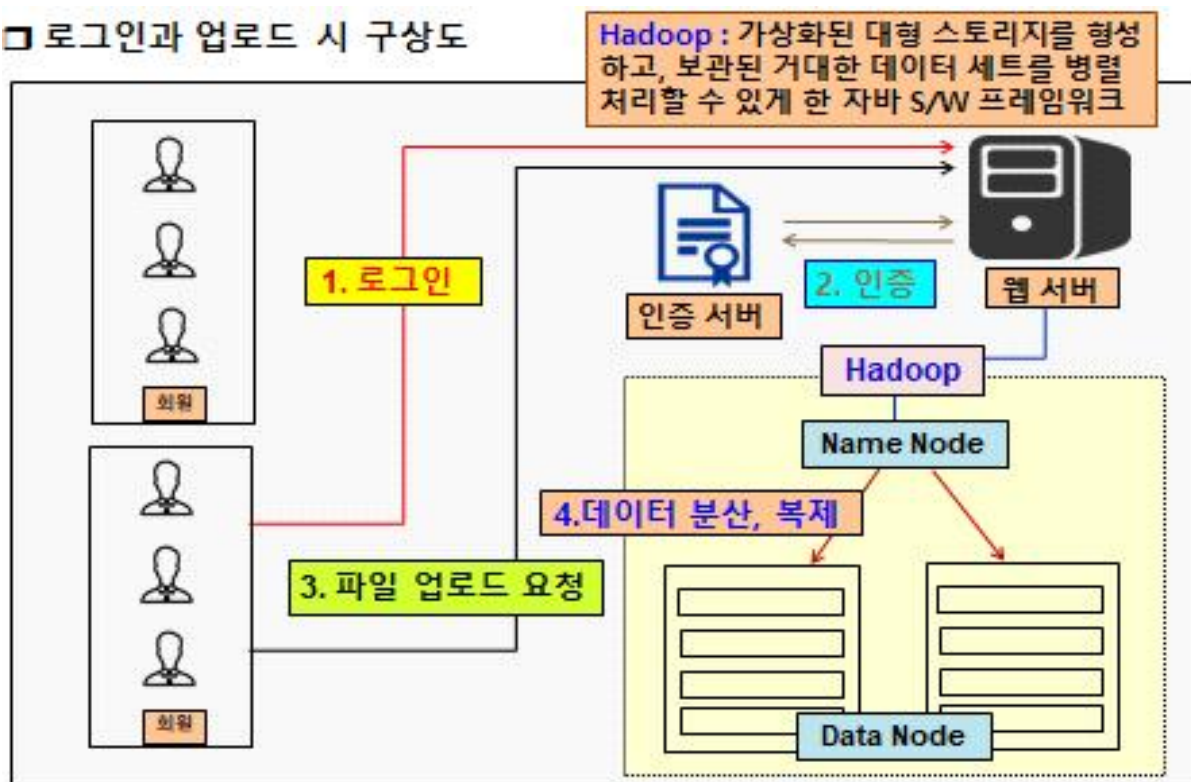
로그인 후, 회원이 파일 업로드 요청을 하면 웹서버를 통해 하둡에 있는 데이터노드 서버에 데이터가 분산, 복제되어 저장되고 다운로드 요청을 하게되면 데이터 노드 서버에 분산 되어있는 데이터를 수집하여 웹서버에 전송 하고 회원에게 파일이 전송하게 된다.

3.2.2 인증시스템 구상도

인증절차는 사용자가 인증서의 생성을 인증서버에 요청하면 웹서버로 해당 사용자에게 대한 인증서를 발급해준다.

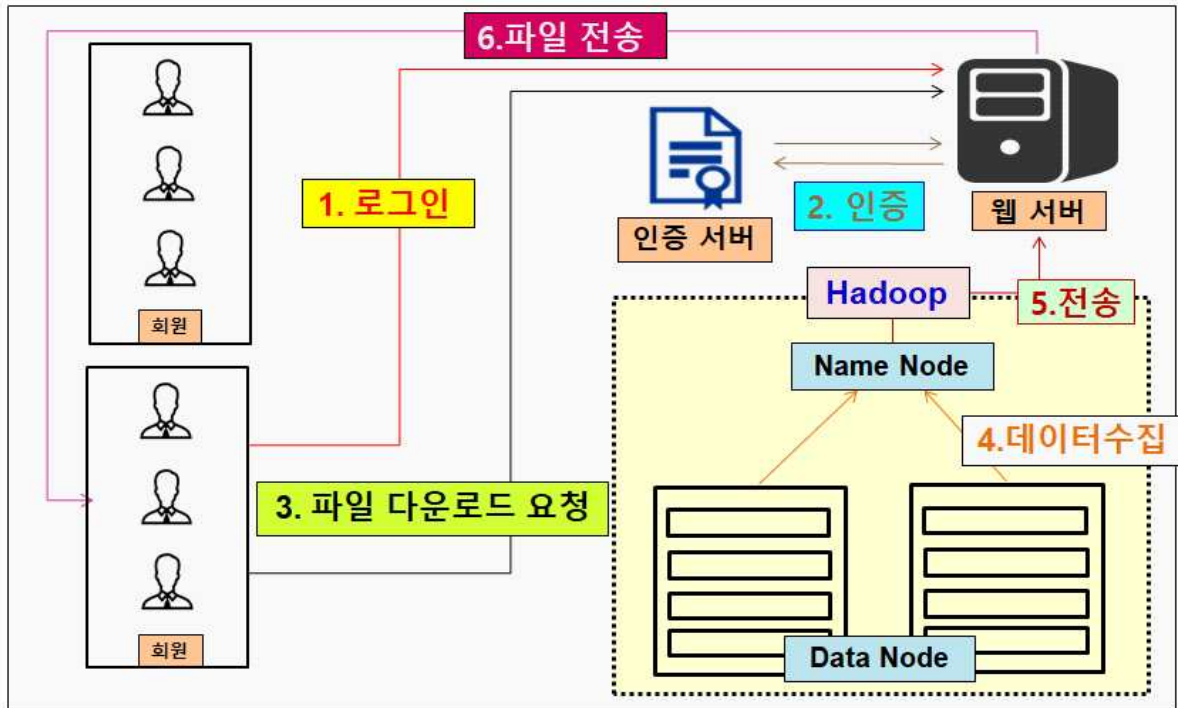
사용자가 해당 웹사이트에 로그인을 시도할 때 발급받은 인증서로 로그인이 가능하기 때문에 인증서버로 해당사용자의 인증서 검증을 요청해 인증서버에서 전자서명으로 검증을 한 후 인증서 시스템 세션에 저장한다. 인증이 완료된 후에는 인증/전자서명 검증 세션을 웹서버로 전송해 사용자의 로그인이 성공 된다.

□ 로그인과 업로드 시 구상도



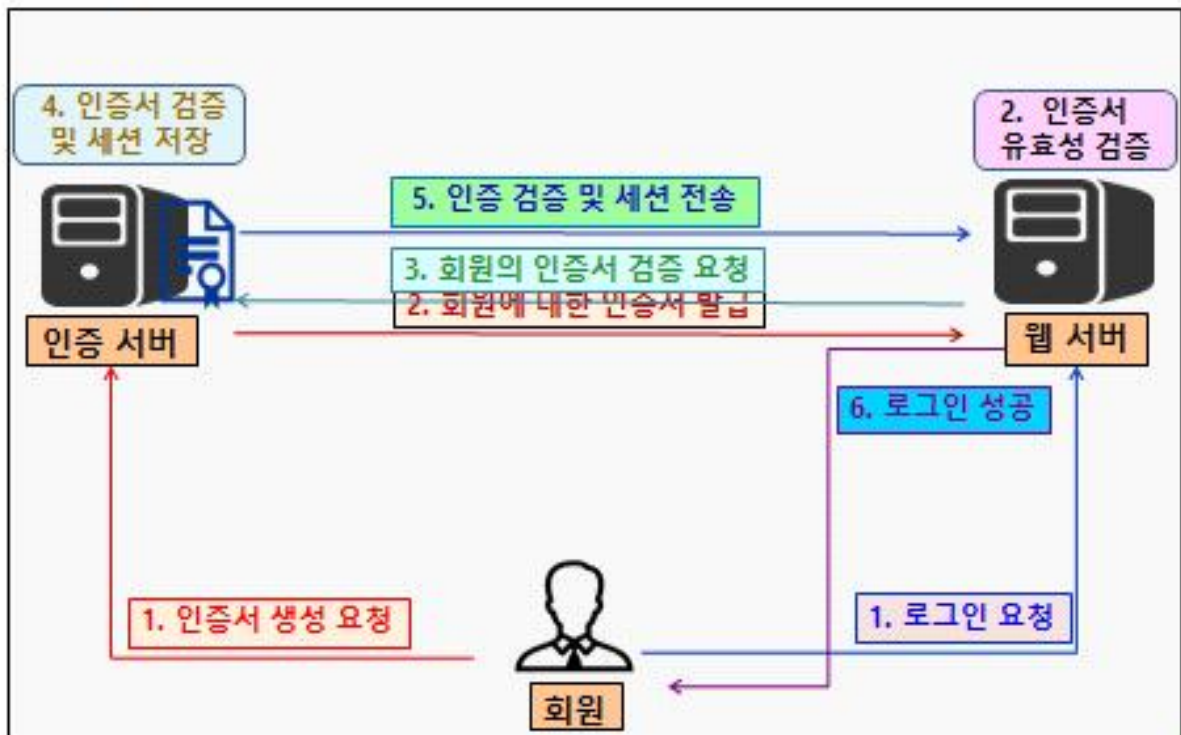
[그림 2. 업로드시 구상도]

□ 로그인과 다운로드 시 구상도



[그림 3. 다운로드시 구상도]

□ 인증절차 구상도

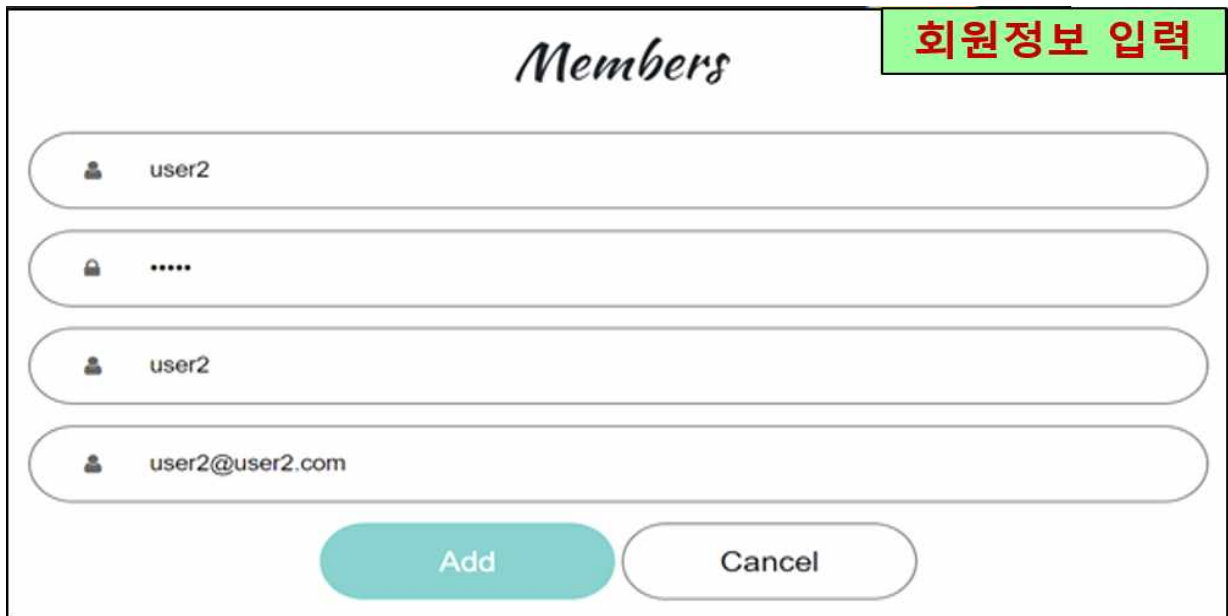


[그림 4. 인증절차 구상도]

3.3 프로그램 구성

3.3.1 웹페이지 기능

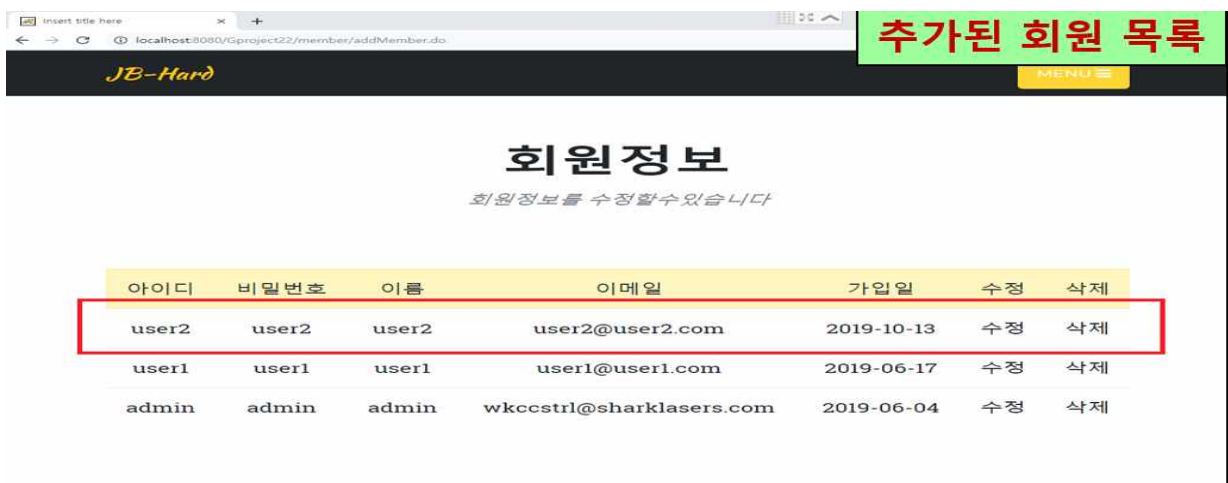
관리자는 사용자 추가 및 삭제로 회원 관리를 할 수 있다. 관리자는 인증서버에 사용자들을 등록한 후 웹서버에도 사용자들의 비밀번호와 이름 이메일을 추가하여 배급한다. 사용자를 추가한 후 언제든지 회원 정보를 수정 및 삭제할 수 있다.



The image shows a web form titled "Members" for adding a new user. It features four input fields: a name field containing "user2", a password field with masked characters ".....", a confirmation name field containing "user2", and an email field containing "user2@user2.com". Below the fields are two buttons: a teal "Add" button and a white "Cancel" button. A green box in the top right corner contains the text "회원정보 입력" (Input member information).

[그림 5. 사용자 등록]

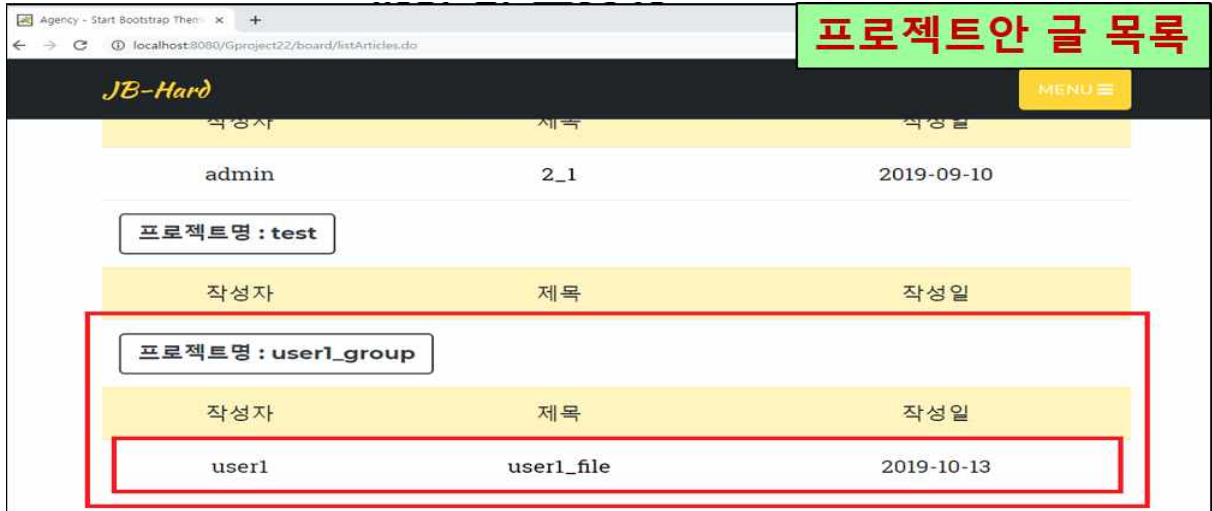
사용자들은 회사에서 사용할 프로젝트를 추가하여 그 프로젝트 안에서 파일 또는 글들을 추가하여 같은 프로젝트를 하는 사용자들과 공유할 수 있다. 이 안에서는 프로젝트와 글은 추가 및 삭제, 수정이 가능하다.



The image is a screenshot of a web browser showing a page titled "회원정보" (Member Information). The page header includes the logo "JB-Hard" and a "MENU" button. Below the title, there is a table listing members. The first row of the table is highlighted with a red border. A green box in the top right corner of the screenshot contains the text "추가된 회원 목록" (List of added members).

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user2	user2	user2	user2@user2.com	2019-10-13	수정	삭제
user1	user1	user1	user1@user1.com	2019-06-17	수정	삭제
admin	admin	admin	wkccstrl@sharklasers.com	2019-06-04	수정	삭제

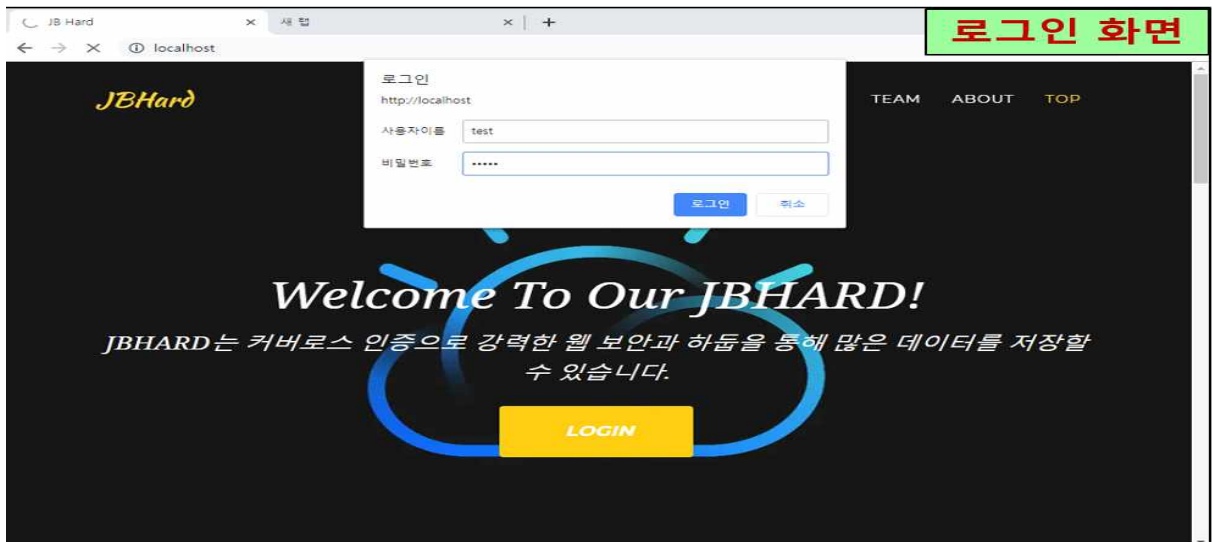
[그림 6. 사용자 수정 및 삭제]



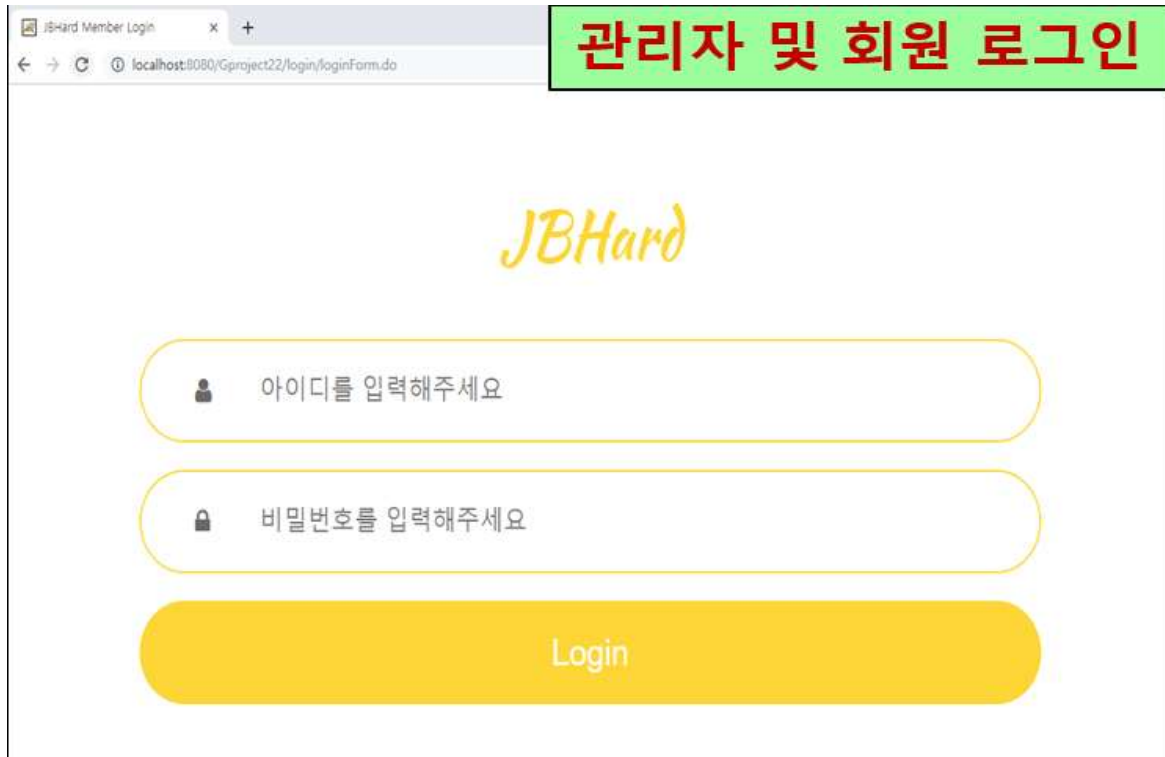
[그림 7. 프로젝트]

3.3.2 인증시스템 기능

관리자가 KDC 서버에 사용자를 추가하여 사용자들에게 아이디와 비밀번호를 발급한다. 그 발급받은 아이디와 비밀번호로 인증 로그인을 하여 사용자를 인증하고 이 사용자가 무슨 직급을 가지고 있는지를 알기 위해 한 번 더 웹 서버에서 로그인을 하게된다.



[그림 8. 인증 로그인]



[그림 9. 직급 로그인]

4. 결론

하둡서버 및 웹서버를 연동하여 사용자 및 그룹 파일 드라이브를 개발/구축 하는데 성공하였으며 로그인 과정에서 Kerberos 인증 방식을 사용해 KDC에서 발급한 Ticket으로 사용자를 검증하는 인증체제로 시스템의 보안성을 확보하였다. 또한 모든 조원들이 업무를 적절히 분담하여 맡은 부분을 완성함에 따라 프로그래밍 능력 및 시스템 구축 능력을 향상시키는 계기가 되었다. 아울러 우리만의 JB-HARD라는 클라우드 시스템을 구축하면서 자그마한 소 회사에서도 중요한 기술, 문서들과 같은 데이터들의 중요성을 알게 되었고 안전하게 보호해야 한다는 생각이 들었다. 더불어 물리적인 보안 등과 같은 것이 추가가 된다면 더욱 보안적으로 완벽한 드라이브가 될 것 이다.

5. 별첨

5.1 발표 PPT

5.2 소스코드

클라우드 시스템 및 인증체제 구현

2019. 10. 29

지도 교수 : 양환석 교수님

클라우드 캐슬
(강지호, 최진석, 양유리, 오수빈)

1

목 차

- 01 조원 편성
- 02 주제 선정
- 03 구 상 도
- 04 추진 경과
- 05 개발 환경 및 개발내용
- 06 개발 시스템 운영
- 07 결론 및 기대효과

2

조원 편성

성명	담당
강지호(조장)	하둡 서버 구축 및 웹 연동, 백엔드 개발, DB설계
최진석	인증서버 구축 및 웹 연동, 보고서 작성
양유리	인증서버 구축 및 웹 연동, 보고서 작성
오수빈	인증서버 구축 및 웹 연동, 보고서 작성
공통	프론트엔드 개발, 웹 서버 구축, ppt작성

3

주제 선정

□ 클라우드 필수 인프라, ICT 업계 경쟁 치열

"클라우드는 기업혁신 필수 인프라, 데이터 활용 방식이 세계 바꿀 것"

클라우드 필수 인프라, ICT 업계 경쟁 치열

클라우드 시장 잡아라 · ICT업계 대격돌

블루오션 '클라우드' 시장 향한 기업 진출 ↑
보안·높은 서비스 품질 강조한 플랫폼 론칭
클라우드 리딩 기업 두고 ICT사 경쟁 치열

최근 기업들의 클라우드 전환이 증가하며, 리딩 기업을 향한 업계 경쟁도 치열해지고 있다. 각 사별로 보안과 높은 서비스 품질 등의 장점을 내세우며 시장 선전에 만장이다.

클라우드인 데이터를 인터넷과 연결된 중앙컴퓨터에 저장, 언제 어디서든 인터넷에 접속해 관련 데이터를 이용할 수 있는 서비스를 말한다. 기업이 클라우드로 전환할 경우 IT 인프라 구축 비용 및 운영비용 절감 등과 IT신기술을 R&D(연구개발), 생산, 마케팅 등 경영 전반에 빠르게 적용할 수 있는 장점이 있다.

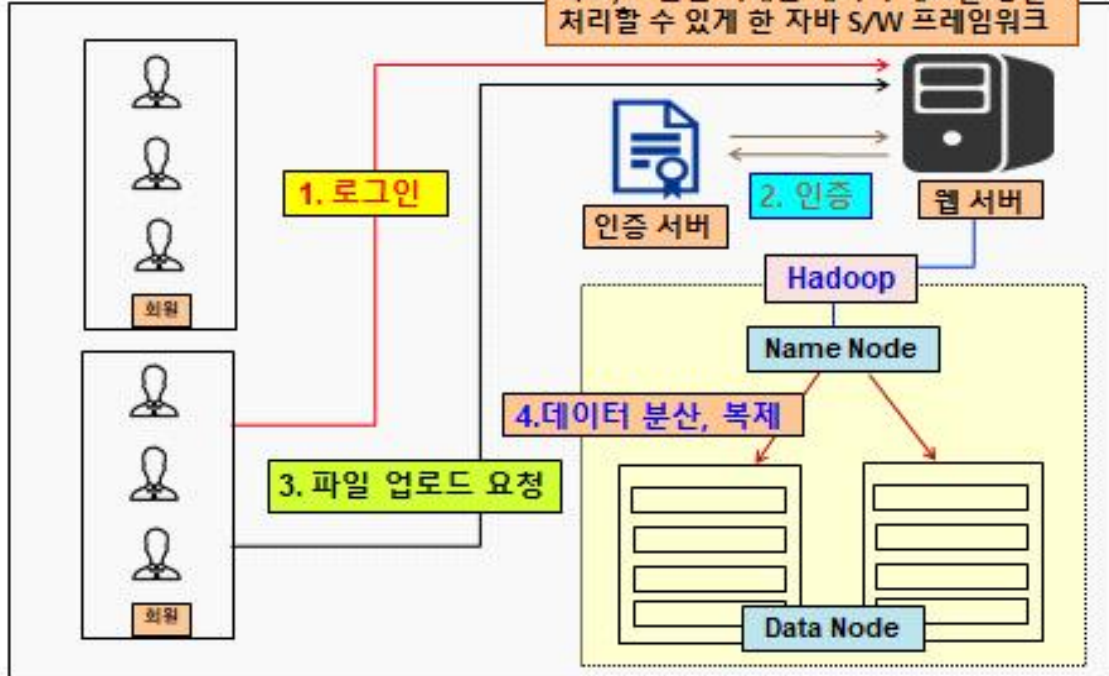
보안관리체제가 완비된 클라우드 시스템 구축방안 연구

4

구상도(1/3)

로그인과 업로드 시 구상도

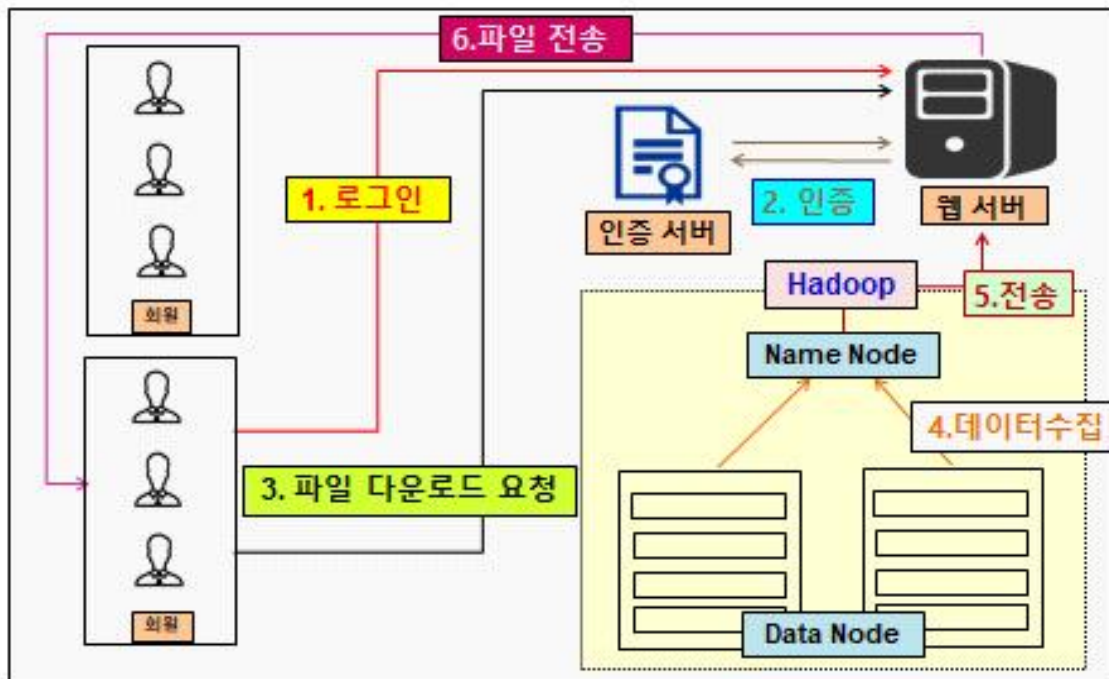
Hadoop : 가상화된 대형 스토리지를 형성하고, 보관된 거대한 데이터 세트를 병렬 처리할 수 있게 한 자바 S/W 프레임워크



5

구상도(2/3)

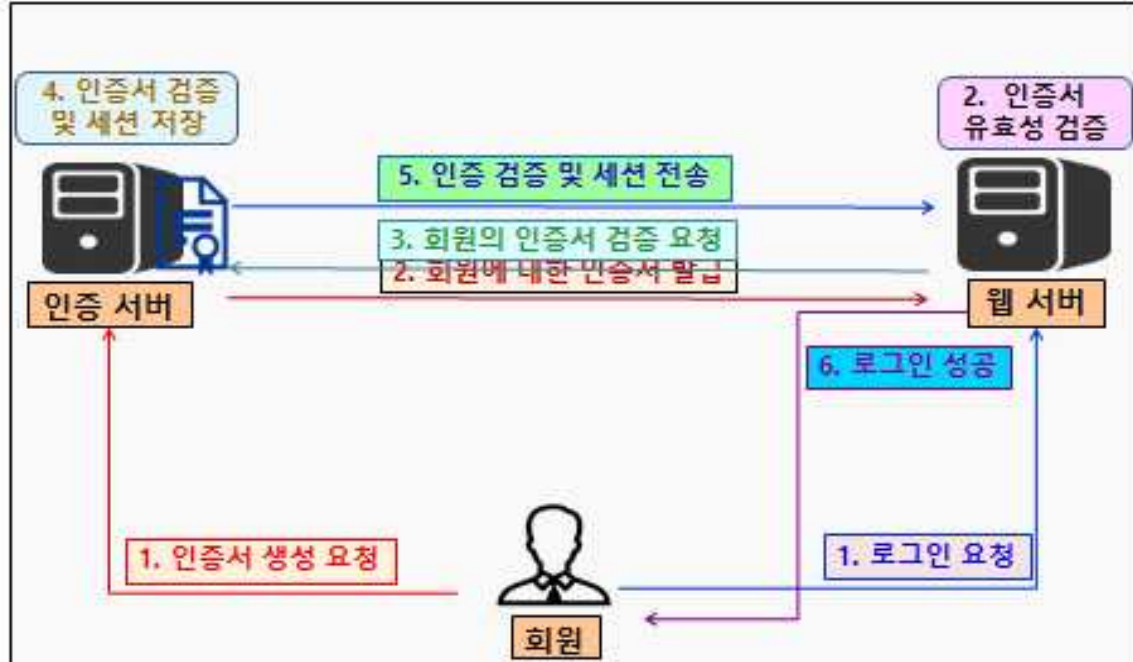
로그인과 다운로드 시 구상도



6

구상도(3/3)

□ 인증절차 구상도



7

추진 경과

추진 업무	추진 기간 (2019년)							
	3월	4월	5월	6월	7월	8월	9월	10월
자료조사/분석	■							
구상도 설계		■						
웹 서버 구축 및 웹 백엔드 개발		■						
하둠 서버 구축 및 웹 연동				■				
인증 서버 구축 및 웹 연동						■		
웹 프론트엔드 개발 및 최종 확인								■

8

개발 환경 및 개발 내용(1/8)

개발 환경



서버운영체제

CentOS 7(Linux)
CentOS 6(Linux)



개발 언어

JSP
HTML/CSS
JAVA



DB

Oracle Database



인증 시스템

Kerberos

9

개발 환경 및 개발 내용(2/8)

회원 관리

```

public void addMember(MemberVO m) {
    try {
        conn
        Strin
        Strin
        Strin
        Strin
        Strin
        Strin
        System
        pstmt
        pstmt
        pstmt
        pstmt
        pstmt
        pstmt
        pstmt
        conn.close();
    } catch (Exception e) {
        // TODO: Handle exception
        e.printStackTrace();
    }
}
    
```

회원 추가

```

public void delMember(String id) {
    try {
        conn
        pstmt
        pstmt
        pstmt
        pstmt
        pstmt
        pstmt
        conn.close();
    } catch (Exception e) {
        // TODO: Handle exception
        e.printStackTrace();
    }
}
    
```

회원 삭제

```

public void modMember(MemberVO memberVO) {
    String id = memberVO.getId();
    String pwd = memberVO.getPwd();
    String name = memberVO.getName();
    String email = memberVO.getEmail();
    try {
        conn = dataFactory.getConnection();
        String query = "update t_member2 set pwd=?,name=?,email=? where id=?";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, pwd);
        pstmt.setString(2, name);
        pstmt.setString(3, email);
        pstmt.setString(4, id);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (Exception e) {
        // TODO: Handle exception
        e.printStackTrace();
    }
}
    
```

회원정보 수정

회원 추가, 삭제, 정보수정 등 관리자의 회원 관리 지원

10

개발 환경 및 개발 내용(3/8)

프로젝트 관리

```
public void addtitleMem(int articleNO, List<MemberVO> mem
    try {
        conn = dataFactory.getConnection();
    }
}

public void deleteArticle(int articleNO) {
    try {
        conn = dataFactory.getConnection();
    }
}

public void updateArticle(ArticleVO article) {
    int articleNO = article.getArticleNO();
    String title = article.getTitle();
    String content = article.getContent();
    String fileName = article.getFileName();

    try {
        conn = dataFactory.getConnection();
        String query = "update t_board2 set title=?, content=?";
        if(fileName != null && fileName.length() != 0) {
            query += ", fileName=?";
        }
        query += " where articleNO=?";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, title);
        pstmt.setString(2, content);
        if(fileName != null && fileName.length() != 0) {
            pstmt.setString(3, fileName);
        }
        pstmt.setInt(4, articleNO);
    }
}
}
```

프로젝트 추가

프로젝트 삭제

프로젝트 수정

프로젝트 추가, 삭제, 정보수정 등 관리자의 프로젝트 관리 지원

11

개발 환경 및 개발 내용(4/8)

회원 권한 관리

```
public List<ArticleVO> listArticles(String id) {
    List<ArticleVO> articlesList = boardDAO.selectAllArticles(id);
    List<ArticleVO> memList = boardDAO.selectAllMembers(id);
    List<ArticleVO> resultList = new ArrayList();
    ArticleVO memVO = null;
    int articleNO, parentNO;
    for(int i=0; i<articlesList.size(); i++) {
        ArticleVO articleVO = articlesList.get(i);
        articleNO = articleVO.getArticleNO();
        parentNO = articleVO.getParentNO();
        for(int j=0; j<memList.size(); j++) {
            memVO = memList.get(j);
            if(articleNO == memVO.getArticleNO()) {
                resultList.add(articleVO);
            }
        }
    }
    return resultList;
}

public List listMembers() {
    List<MemberVO> membersList = new ArrayList();

    try {
        conn = dataFactory.getConnection();
        String query = "select * from t_member2 order by joinDate desc";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            MemberVO vo = new MemberVO();
            String id = rs.getString("id");
            String pwd = rs.getString("pwd");
            String name = rs.getString("name");
            String email = rs.getString("email");
            Date joinDate = rs.getDate("joinDate");
            MemberVO memberVO = new MemberVO(id, pwd, name, email, joinDate);
            membersList.add(memberVO);
        }
    }
}

회원 및 관리자 로그인 시 정보를 획득, 권한 관리에 활용
```

일반회원 정보 획득

관리자 정보 획득

12

개발 환경 및 개발 내용(5/8)

프로젝트 내의 글 관리

```

public int insertNewArticle(ArticleVO article) {
    int articleNO = getNewArticleNo();
    try {
        conn = dataFactory.getConnection();
        int parentNO = article.getParentNo();
        String title = article.getTitle();
        String content = article.getContent();
        String id = article.getId();
        System.out.println("filename =====");
        String filename = article.getFileName();
        String query = "INSERT INTO t_board2 (parentno, title, content, filename, id)
            VALUES (?, ?, ?, ?, ?)";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.setInt(2, parentNO);
        pstmt.setString(3, title);
        pstmt.setString(4, content);
        pstmt.setString(5, filename);
        pstmt.setString(6, id);
        pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
    
```

파일/글 추가

```

public void deleteArticle(int articleNO) {
    try {
        conn = dataFactory.getConnection();
        String query = "DELETE FROM t_board2 WHERE articleNO=?";
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
    
```

파일/글 삭제

```

public void updateArticle(ArticleVO article) {
    int articleNO = article.getArticleNO();
    String title = article.getTitle();
    String content = article.getContent();
    String filename = article.getFileName();
    try {
        conn = dataFactory.getConnection();
        String query = "UPDATE t_board2 SET title=?, content=? WHERE articleNO=?";
        if(filename != null && filename.length() != 0) {
            query += ", filename=?";
        }
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, title);
        pstmt.setString(2, content);
        if(filename != null && filename.length() != 0) {
            pstmt.setString(3, filename);
        }
        pstmt.executeUpdate();
    } catch (Exception e) {
        e.printStackTrace();
    }
}
    
```

파일/글 수정

관리자와 사용자가 프로젝트 내의 파일/글 관리

13

개발 환경 및 개발 내용(6/8)

KDC(인증서버) 설정(1/2)

```

[root@kerberos ~]# cat /etc/krb5.conf
[logging]
default = FILE:/var/log/krb5libs.log
kdc = FILE:/var/log/krb5kdc.log
admin_server = FILE:/var/log/kadmind.log

[libdefaults]
dns_lookup_functions =
ticket_lifetime = 10h
renew_lifetime = 7d
forwardable = true
rdns = false
default_realm = JBHARD.COM
default_ccache_name = /tmp/cccachetmp

[realms]
JBHARD.COM = {
    kdc = jbhardsvc.kerberos.jbhards.com
    admin_server = jbhardsvc.kerberos.jbhards.com
}

[domain_realm]
.jbhards.com = JBHARD.COM
.jbhards.com = JBHARD.COM
    
```

krb5.conf 파일 수정

※ DNS와 NTP는 기본적으로 설정이 되어야 함

```

[root@kerberos ~]# cat /etc/kadm5.acl
*/admin@JBHARD.COM*
    
```

kadm5.acl 파일 수정

```

[root@kerberos ~]# kdb5_util create -s -r JBHARD.COM
Loading random data
Initializing database '/var/kerberos/krb5kdc/principal' for realm 'JBHARD.COM',
master key name 'K/M@JBHARD.COM'
You will be prompted for the database Master Password.
It is important that you NOT FORGET this password.
Enter KDC database master key:
Re-enter KDC database master key to verify:
    
```

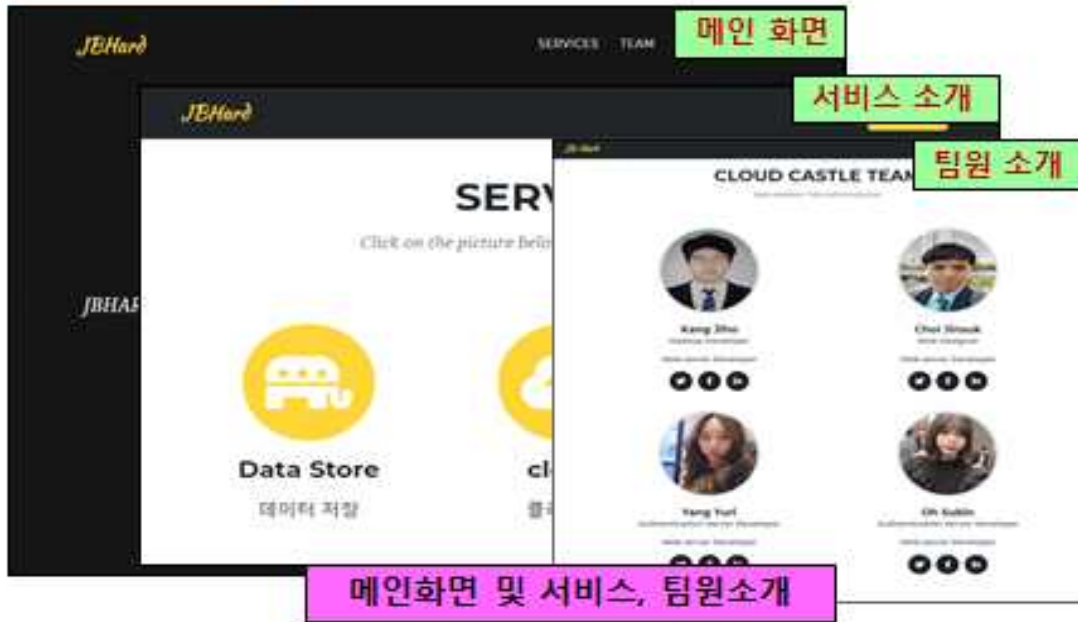
KDC DB master key 등록

도메인 구성 및 설정 파일 수정 후 KDC Database 설정

14

개발 시스템 운영 (1/10)

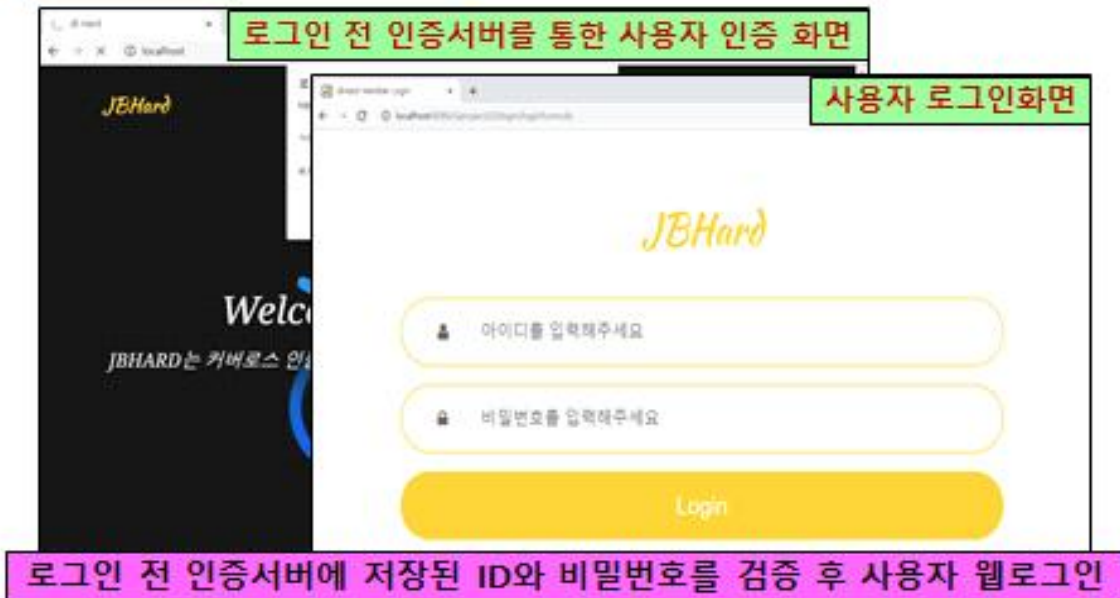
메인 화면



17

개발 시스템 운영 (2/10)

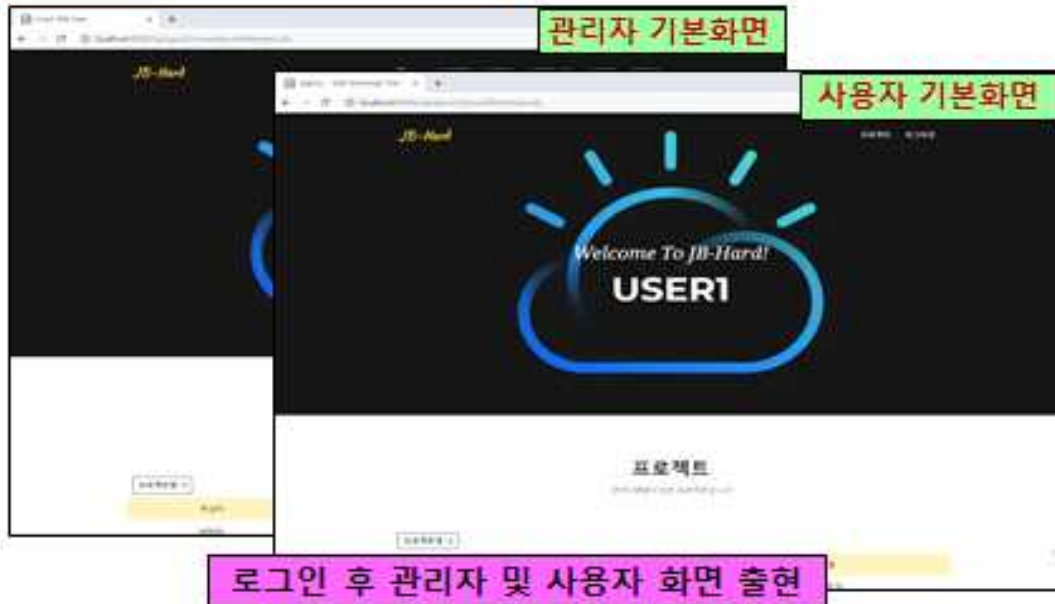
인증 및 로그인 화면



18

개발 시스템 운영 (3/10)

관리자 및 사용자 화면



19

개발 시스템 운영-관리자 (4/10)

회원추가



20

개발 시스템 운영-관리자 (5/10)

회원삭제

회원 삭제

삭제 결과창

삭제 후 회원 목록

회원정보

회원정보를 수정할수있습니다

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user1	user1	서치후	user1@joongbu.ac.kr	2019-06-17	수정	삭제
admin	admin	관리자	wkccst1@sharklazers.com	2019-06-04	수정	삭제

회원 삭제 버튼 클릭 시 회원 삭제

21

개발 시스템 운영-관리자 (6/10)

회원수정

회원 수정

회원정보 수정 입력

수정 후 회원 목록

회원정보

회원정보를 수정할수있습니다

아이디	비밀번호	이름	이메일	가입일	수정	삭제
user1	user1	person	user1@joongbu.ac.kr	2019-06-17	수정	삭제
admin	admin	관리자	wkccst1@sharklazers.com	2019-06-04	수정	삭제

회원 비밀번호, 이름 및 이메일 변경 가능

22

개발 시스템 운영-관리자 (7/10)

프로젝트 추가

The screenshot shows the 'Project Add' form in the JB-Hard application. The form includes fields for '이름' (Name), '설명' (Description), '담당자' (Responsible Person), and '그룹' (Group). A '프로젝트명 : user1_group' is entered in the Group field. A '추가' (Add) button is visible. Below the form, a table lists project details with columns for '이름', '설명', and '작성일' (Creation Date). A terminal window at the bottom shows the command 'hadoop dfs -ls /user/hive/warehouse/11' and its output, listing a file 'joongbu.jpg' with permissions '-rw-r--r--' and size '3'.

프로젝트 추가

회원 지정 및 파일 저장

추가 후 프로젝트 목록

추가 후 하둡에 파일 저장

```
hadoop@wikibooks01 [바탕화면] $ hadoop dfs -ls /user/hive/warehouse/11
Warning: SHADOOP_HOME is deprecated.

Found 1 items
-rw-r--r--  3 hive supergroup      19621 2019-06-27 23:16 /user/hive/warehouse/11/joongbu.jpg
hadoop@wikib
```

프로젝트 사용 회원 및 파일 저장 시 하둡에 저장

23

개발 시스템 운영-사용자 (8/10)

회원 로그인

The screenshot shows the '회원 로그인' (Member Login) page. The login form includes fields for '아이디' (ID) and '비밀번호' (Password). A '로그인' (Login) button is present. Below the login form, a list of projects is shown with '프로젝트 내용 1' and '프로젝트 내용 2' visible. The '로그인' button is highlighted with a red box. Below the project list, the Joongbu University logo is displayed, featuring the text 'JOONGBU UNIVERSITY' and '중부대학교' (Joongbu University) around a central emblem. Below the logo, the text '파일명 : joongbu.jpg' and '파일 내려받기' (Download File) is visible.

회원 로그인 시 프로젝트 목록

프로젝트 내용 1

프로젝트 내용 2

회원 로그인 시 참여 프로젝트 목록 및 프로젝트 내용 열람 가능

24

개발 시스템 운영-사용자 (9/10)

프로젝트 삭제

프로젝트 삭제

삭제 버튼 클릭

삭제 후 프로젝트 목록

이름	버전	작성일
admin	1.1	2019-09-30
프로젝트명 : user_together2		
작성자	재혁	작성일
admin	2.1	2019-09-30
프로젝트명 : test		
작성자	재혁	작성일
프로젝트명 : user1_group		
작성자	재혁	작성일

프로젝트 삭제 시 하위 파일 및 글도 함께 삭제

25

개발 시스템 운영-사용자 (10/10)

프로젝트안 파일 및 글추가

파일/글 추가할 프로젝트

추가 버튼 클릭

추가할 파일/글 입력

프로젝트안 글 목록

```
(hadoop@wikibooks0) 바탕화면$ hadoop dfs -ls /user/h  
Warning: SHADOOP_HOME is deprecated.  
  
Found 1 items  
-rw-r--r-- 3 hive se/12/joongbu.jpg
```

파일/글 추가 후 하둡에 파일 저장

파일 및 글 추가시 하둡에 파일 저장

26

결론 및 기대효과



◆ 결 론

- 하둡 서버 및 웹 서버를 연동하여 사용자 그룹 파일 드라이브 웹 사이트를 개발/구축하는데 성공
- 로그인 과정에서 Kerberos 인증 방식을 사용하여 KDC에서 발급한 ticket으로 사용자를 검증하는 인증 체제로 시스템의 보안성을 확보

◆ 기대효과

- 모든 조원들이 업무를 적절히 분담하여 맡은 부분을 완성함에 따라 프로그래밍 능력 및 시스템 구축 능력을 향상시키는 계기
- 아울러 작품 완성을 통해 클라우드 시스템과 보안 관리의 중요성을 재인식

-끝-

27

Q & A

Thank you

28


```

        try {
            Class.forName("org.apache.hive.jdbc.HiveDriver");
            Connection conn
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
            Statement stmt = conn.createStatement();

            String hdfs_dir="/user/hive/warehouse/"+articleNO;
            String local_dir="/var/server_share";//Integer.toString(articleNO);

            String sql = "dfs -get "+hdfs_dir+" "+local_dir;
            stmt.executeQuery(sql);
            System.out.println("server에 파일추가 완료");

        } catch (ClassNotFoundException e) {
            System.out.println("class.forName실패");
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("connection실패");
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public void rmrdfs(int articleNO) {
        try {
            Class.forName("org.apache.hive.jdbc.HiveDriver");
            Connection conn
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
            Statement stmt = conn.createStatement();

            String hdfs_dir="/user/hive/warehouse/"+articleNO;

            String sql = "dfs -rmr "+hdfs_dir;
            stmt.executeQuery(sql);
            System.out.println("hadoop에서 파일삭제 완료");

        } catch (ClassNotFoundException e) {
            System.out.println("class.forName실패");
            // TODO Auto-generated catch block
            e.printStackTrace();
        } catch (SQLException e) {
            System.out.println("connection실패");
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public void rmrfiledfs(int articleNO,String filename) {
        try {
            Class.forName("org.apache.hive.jdbc.HiveDriver");
            Connection conn
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
            Statement stmt = conn.createStatement();

            String hdfs_file="/user/hive/warehouse/"+articleNO+"/"+filename;

```

```
String sql = "dfs -rmr "+hdfs_file;
stmt.executeQuery(sql);
System.out.println("hadoop에서 파일삭제 완료");

    } catch (ClassNotFoundException e) {
        System.out.println("class.forName실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (SQLException e) {
        System.out.println("connection실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}
```

articleForm.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" isELIgnored="false"%>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%
    request.setCharacterEncoding("utf-8");
%>
    <c:set var="contextPath" value="${pageContext.request.contextPath }"/>
<!DOCTYPE html>
<html>
<head>
<style>
.write {
    font-size: 40px;
    text-align:center;
}
.page-section {
padding: 100px 0;
}

form {
margin: 0 auto;
width: 900px;
height: 500px;
padding: 2em;
border: 1px solid #CCC;
border-radius: 1em;
}
form div + div {
margin-top: 1em;
}
label {
display: inline-block;
width: 100px;
text-align:left;
}
input,textarea {
font : 1em sans-serif;
width: 400px;
-moz-box-sizing:border-box;
```

```

        box-sizing: border-box;
        border: 1px solid #999;
    }
    textarea {
        vertical-align: top;
        height: 5em;
        resize: vertical;
    }
    #msg {
        height: 176px;
    }
    .button {
        padding-left: 90px;
    }
    button {
        margin-left: .5em;
    }

</style>
<meta charset="UTF-8">
<link href="{contextPath }/page/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<link href="{contextPath }/page/css/agency.css" rel="stylesheet">
<!-- Custom fonts for this template -->
<link href="{contextPath }/page/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet"
type="text/css">
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Droid+Serif:400,700,400italic,700italic'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700'
rel='stylesheet' type='text/css'>
<title>Insert title here</title>
<script src="http://code.jquery.com/jquery-latest.min.js"></script>
<script type="text/javascript">
    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();
            reader.onload = function (e) {
                $('#preview').attr('src', e.target.result);
            }
            reader.readAsDataURL(input.files[0]);
        }
    }

    function backToList(obj){
        obj.action="{contextPath}/member/listMembers.do";
        obj.submit();
    }
</script>
</head>
<body>
    <section class="page-section" id="write">
        <div class="write" >Project Add</div>
        <form name="articleForm" method="post" action="{contextPath

```

```

}/board/addArticle.do" enctype="multipart/form-data">

    <div>
        <label for="name">Title:</label>
        <input type="text" id="title" name="title"/></td>
    </div>
    <div>
        <label for="content">Content:</label>
        <textarea name="content" id="msg"></textarea>
    </div>
    <div class="form-group">
        <label for="File">File Upload</label>
        <input type="file" id="File" name="FileName"/>
    </div>

    <div class="checkbox">
        <label for="name">Members</label>
        <c:forEach var="mem" items="{membersList }"
varStatus="memNum">
            <c:if test="{(memNum.count%4)==0 }">
                <br>
            </c:if>
            <input type="checkbox" name="mem"
value="{mem.id }"/>{mem.id }
        </c:forEach>
    </div>
    <div class="btn">
        <input type="submit" class="btn btn-primary btn-sm"
value="Add"/>
        <input type="button" class="btn btn-default btn-sm"
value="List" onClick="backToList(this.form)"/>
    </div>
</form>
</section>
</body>
</html>

```

ArticleVO

```

package brd;

import java.sql.Date;

public class ArticleVO {
    private int level;
    private int articleNO;
    private int parentNO;
    private String title;
    private String content;
    private String fileName;

    public String getFileName() {
        return fileName;
    }

    public void setFileName(String fileName) {

```

```

        this.fileName = fileName;
    }

    private String id;
    private Date writeDate;

    public ArticleVO() {
    }

    public int getLevel() {
        return level;
    }

    public void setLevel(int level) {
        this.level = level;
    }

    public int getArticleNO() {
        return articleNO;
    }

    public void setArticleNO(int articleNO) {
        this.articleNO = articleNO;
    }

    public int getParentNO() {
        return parentNO;
    }

    public void setParentNO(int parentNO) {
        this.parentNO = parentNO;
    }

    public String getTitle() {
        return title;
    }

    public void setTitle(String title) {
        this.title = title;
    }

    public String getContent() {
        return content;
    }

    public void setContent(String content) {
        this.content = content;
    }

    public String getId() {
        return id;
    }

    public void setId(String id) {
        this.id = id;
    }

    public Date getWriteDate() {
        return writeDate;
    }

    public ArticleVO(int level,int articleNO,int parentNO,String title,String content,String

```

```

FileName,String id) {
    super();
    this.level = level;
    this.articleNO = articleNO;
    this.parentNO = parentNO;
    this.title = title;
    this.content = content;
    this.fileName = FileName;
    this.id = id;
}
public void setWriteDate(Date writeDate) {
    this.writeDate = writeDate;
}
}

```

BoardController

```

package brd;

import java.io.File;

import java.io.IOException;
import java.io.PrintWriter;
import java.nio.file.Paths;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import org.apache.commons.fileupload.FileItem;
import org.apache.commons.fileupload.FileUploadException;
import org.apache.commons.fileupload.disk.DiskFileItemFactory;
import org.apache.commons.fileupload.servlet.ServletFileUpload;
import org.apache.commons.io.FileUtils;

import mem.MemberDAO;
import mem.MemberVO;

@WebServlet("/board/*")
public class BoardController extends HttpServlet{
    //private static String ARTICLE_IMAGE_REPO = "C:\\www\\myJSP\\wwwboard\\wwwG_image";
    private static String ARTICLE_IMAGE_REPO = File.separator + "dir";
    /**
     *
     */
    private static final long serialVersionUID = 1L;
    BoardService boardService;
    ArticleVO articleVO;
    MemberDAO memberDAO;
}

```



```

List<MemberVO> membersList = new ArrayList();

public void init(ServletConfig config) throws ServletException{
    super.init(config);
    boardService = new BoardService();
    articleVO = new ArticleVO();
    memberDAO = new MemberDAO();
}

protected void doGet(HttpServletRequest request,HttpServletResponse response)
throws ServletException,IOException{
    doHandle(request,response);
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    doHandle(request,response);
}

private void doHandle(HttpServletRequest request,HttpServletResponse response)
throws ServletException,IOException{
    //System.out.println(Paths.get("1", "2"));
    String nextPage = "";
    request.setCharacterEncoding("utf-8");
    response.setContentType("text/html;charset=utf-8");
    HttpSession session = request.getSession();
    System.out.println("request.getRequestURL() = "+request.getRequestURL());
    System.out.println("request.getPathInfo() = "+request.getPathInfo());
    String action = request.getPathInfo();
    System.out.println("action:"+action);

    if(session.getAttribute("id")==null) {
        PrintWriter pw = response.getWriter();
        pw.write("<script>                                alert('로그인해주세요');
location.href='"+request.getContextPath()+"/login/loginForm.do';</script>");
        return;
    }
    try {
        System.out.println("nextPage 1 ===== "+nextPage);
        List<ArticleVO> articlesList = new ArrayList<ArticleVO>();
        if((action==null && session.getAttribute("id").equals("admin")) ||
action.equals("/listArticles.do")) {
            String id = (String) session.getAttribute("id");
            System.out.println("id = ===== "+id);
            if(id.equals("admin")) {
                articlesList = boardService.listAllArticles();
            }else {
                articlesList = boardService.listArticles(id);
            }
        }

        System.out.println("aaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaaa");
        request.setAttribute("articlesList",articlesList);
        nextPage = "/board2/listArticles.jsp";

        }else if(action.equals("/articleForm.do")) {
            List<MemberVO> membersList = new ArrayList();

```

```

        membersList=boardService.findmembers();
        request.setAttribute("membersList", membersList);
        nextPage = "/board2/articleForm.jsp";
    }else if(action.equals("/addArticle.do")) {
        /*
        String[] mem = request.getParameterValues("mem");
        for(String memid : mem) {
            System.out.println("memid ======"+memid);
        }
        */
        int articleNO = 0;
        Map<String,String> articleMap =
upload(request,response);

        String title = articleMap.get("title");
        String content = articleMap.get("content");
        String FileName = articleMap.get("FileName");

        for(MemberVO id : membersList) {
            System.out.println("idididididididididid ===== "+id.getId());
        }
        //System.out.println("mem[1]="+articleMap.get("mem"));

        articleVO.setParentNO(0);
        articleVO.setId((String)session.getAttribute("id"));
        articleVO.setTitle(title);
        articleVO.setContent(content);
        articleVO.setFileName(FileName);
        articleNO= boardService.addArticle(articleVO,membersList);

if(FileName!=null && FileName.length()!=0) {
    //File srcFile = new File(ARTICLE_IMAGE_REPO+"ww"+"temp"+"ww"+FileName);
File srcFile = new File(ARTICLE_IMAGE_REPO+File.separator
+"temp"+File.separator+FileName);

    //File destDir = new File(ARTICLE_IMAGE_REPO+"ww"+articleNO);
File destDir = new File(ARTICLE_IMAGE_REPO+File.separator +articleNO);
destDir.mkdirs();

    FileUtils.moveToDirectory(srcFile, destDir, true);
String dir_name=ARTICLE_IMAGE_REPO+File.separator +articleNO;

    boardService.intohdfs(dir_name, articleNO, FileName);

        File[] deleteList = destDir.listFiles();
        for(int i = 0;i<deleteList.length;i++) {
            System.out.println("i값 = "+i+" 삭제 결과 = "+deleteList[i].delete());
        }
System.out.println("폴더명 = "+destDir.getName()+" 폴더 삭제 결과 = "+destDir.delete());
    }
        PrintWriter pw = response.getWriter();
        pw.print("<script>" + "alert('프로젝트를 추가했습니다.');"
+ "location.href='"+request.getContextPath()+"/member/'+\"listMembers.do;\" + "</script>");
System.out.println("id_+_+_+_+="+session.getAttribute("id"));
        return;
    }else if(action.equals("/viewArticle.do")) {

```

```

        String articleNO = request.getParameter("articleNO");
        articleVO
boardService.viewArticle(Integer.parseInt(articleNO));

System.out.println("123456789123456789====="+ articleVO.getFileName());
        request.setAttribute("article", articleVO);
        nextPage = "/board2/viewArticle.jsp";
    }else if(action.equals("/modArticle.do")) {
        Map<String,String>        articleMap
upload(request,response);
        int        articleNO
Integer.parseInt(articleMap.get("articleNO"));
        articleVO.setArticleNO(articleNO);
        String title = articleMap.get("title");
        String content = articleMap.get("content");
        String FileName = articleMap.get("FileName");
        articleVO.setParentNO(0);
        articleVO.setld((String)session.getAttribute("id"));
        articleVO.setTitle(title);
        articleVO.setContent(content);
        articleVO.setFileName(FileName);
        boardService.modArticle(articleVO);

        if(FileName != null && FileName.length() !=0) {
            String originalFileName = articleMap.get("originalFileName");
            //File srcFile = new File(ARTICLE_IMAGE_REPO+ "WW" + "temp" +
"WW" + FileName);
            File srcFile = new File(ARTICLE_IMAGE_REPO+ File.separator +
"temp" + File.separator + FileName);
            //File destDir = new File(ARTICLE_IMAGE_REPO+ "WW" +
articleNO);
            File destDir = new File(ARTICLE_IMAGE_REPO+ File.separator +
articleNO);
            destDir.mkdirs();
            FileUtils.moveFileToDirectory(srcFile, destDir, true);
            //File oldFile = new File(ARTICLE_IMAGE_REPO+"WW"+articleNO+"WW"+originalFileName);
            File oldFile = new File(ARTICLE_IMAGE_REPO+File.separator +articleNO+File.separator
+originalFileName);

boardService.moddfs(ARTICLE_IMAGE_REPO,articleNO,FileName);//oldFile.getAbsolutePath()
            oldFile.delete();
        }
        PrintWriter pw = response.getWriter();
        pw.print("<script>"+ " alert('글을 수정했습니다');" + "
location.href="+request.getContextPath()
+"/board/viewArticle.do?articleNO="+articleNO+"";"+ "</script>");
        return;
    }else if (action.equals("/removeArticle.do")) {
        int articleNO = Integer.parseInt(request.getParameter("articleNO"));
        List<Integer> articleNOList = boardService.removeArticle(articleNO);
        for (int _articleNO : articleNOList) {
            boardService.deletedfs(_articleNO);
            //File imgDir = new File(ARTICLE_IMAGE_REPO + "WW" + _articleNO);
            File imgDir = new File(ARTICLE_IMAGE_REPO + File.separator +
_articleNO);
            if (imgDir.exists()) {

```

```

        FileUtils.deleteDirectory(imgDir);
    }
}
PrintWriter pw = response.getWriter();
if(session.getAttribute("id").equals("admin")) {
    pw.print("<script>" + " alert('글을 삭제했습니다.');" + " location.href='" + request.getContextPath()
        + "/member/listMembers.do';" + "</script>");
}
}else {
    pw.print("<script>" + " alert('글을 삭제했습니다.');" + " location.href='" + request.getContextPath()
        + "/board/listArticles.do';" + "</script>");
}
return;
}
else if(action.equals("/replyForm.do")) {
System.out.println("parentNO=====++++++==" + request.getParameter("parentNO").trim());
int parentNO =
Integer.parseInt(request.getParameter("parentNO").trim());
session = request.getSession();
System.out.println("session=====++++++==" + session);
System.out.println("session id =====++++++=="
"+session.getAttribute("id"));
session.setAttribute("parentNO", parentNO);

PrintWriter pw = response.getWriter();
pw.print("<script> location.href='"
request.getContextPath()+ "/board2/replyForm.jsp';" + "</script>");
return;
}
else if(action.equals("/addReply.do")) {
    session = request.getSession();
    int parentNO = (Integer) session.getAttribute("parentNO");
    session.removeAttribute("parentNO");
    Map<String,String> articleMap =
upload(request,response);

    String title = articleMap.get("title");
    String content = articleMap.get("content");
    String FileName = articleMap.get("FileName");
    articleVO.setParentNO(parentNO);
    articleVO.setld((String)session.getAttribute("id"));
    articleVO.setTitle(title);
    articleVO.setContent(content);
    articleVO.setFileName(FileName);
    int articleNO = boardService.addReply(articleVO);
    if (FileName != null && FileName.length() != 0) {
        //File srcFile = new File(ARTICLE_IMAGE_REPO +
"WW" + "temp" + "WW" + FileName);
        File srcFile = new File(ARTICLE_IMAGE_REPO +
File.separator + "temp" + File.separator + FileName);
        //File destDir = new File(ARTICLE_IMAGE_REPO +
"WW" + articleNO);
        File destDir = new File(ARTICLE_IMAGE_REPO +
File.separator + articleNO);

        destDir.mkdirs();
        FileUtils.moveFileToDirectory(srcFile, destDir, true);
        boardService.intohdfs(ARTICLE_IMAGE_REPO,

```

```

articleNO, FileName);

                                File[] deleteList = destDir.listFiles();
                                for(int i = 0;i<deleteList.length;i++) {
                                    System.out.println("i값 = "+i+" 삭제결과
= "+deleteList[i].delete());
                                }
                                System.out.println("폴더명 = "+destDir.getName()+" 폴더
삭제 결과 = "+destDir.delete());
                                }
                                PrintWriter pw = response.getWriter();
                                pw.print("<script>" + " alert('답글을 추가했습니다.');" + "
location.href=\"" + request.getContextPath()
                                +
"/board/viewArticle.do?articleNO="+articleNO+"\";" + "</script>");
                                return;
                            }

                                RequestDispatcher          dispatch          =
request.getRequestDispatcher(nextPage);
                                dispatch.forward(request, response);
                            } catch (Exception e) {
                                // TODO: handle exception
                                System.out.println("nextPage 3 ===== "+nextPage);
                                System.out.println("request.getRequestURL()
="+request.getRequestURL());
                                System.out.println("request.getPathInfo() = "+request.getPathInfo());
                                e.printStackTrace();
                            }
                        }
                    private Map<String,String> upload(HttpServletRequest request,HttpServletResponse
response){
                        Map<String,String> articleMap = new HashMap<String,String>();
                        //List<MemberVO> membersList = new ArrayList();
                        String encoding = "utf-8";
                        File currentDirPath = new File(ARTICLE_IMAGE_REPO);
                        DiskFileItemFactory factory = new DiskFileItemFactory();
                        factory.setRepository(currentDirPath);
                        factory.setSizeThreshold(1024*1024);
                        ServletFileUpload upload = new ServletFileUpload(factory);
                        try {
                            List items = upload.parseRequest(request);
                            for(int i=0;i<items.size();i++) {
                                FileItem fileItem = (FileItem)items.get(i);
                                if(fileItem.isFormField()) {
                                    if(fileItem.getFieldName().equals("mem")) {
                                        MemberVO vo = new MemberVO();
                                        String id = fileItem.getString(encoding);
                                        vo.setId(id);
                                        membersList.add(vo);
                                    }
                                }
                                System.out.println(fileItem.getFieldName()+" = "+fileItem.getString(encoding));
                                articleMap.put(fileItem.getFieldName(), fileItem.getString(encoding));
                            }
                        } else {
                            System.out.println("파라미터명 : "+fileItem.getFieldName());
                            System.out.println("파일명 : "+fileItem.getName());
                        }
                    }
                }
            }
        }
    }
}

```

```

        System.out.println("파일 크기 : "+fileItem.getSize()+"bytes");
        articleMap.put(fileItem.getFieldName(), fileItem.getName());
        System.out.println("1 파일명 : "+fileItem.getName()+" len :
"+fileItem.getName().length());
        if(fileItem.getSize()>0) {
            int idx = fileItem.getName().lastIndexOf("WW");
            if(idx!=-1) {
                idx=fileItem.getName().lastIndexOf("/");
            }

            String fileName = fileItem.getName().substring(idx+1);
            System.out.println("2 파일명 : "+fileName+" len :
"+fileName.length());
            articleMap.put(fileItem.getFieldName(), fileName);
            //File uploadFile = new File(currentDirPath+"WWtempWW"+fileName);
            File uploadFile = new
File(currentDirPath+File.separator + "temp"+File.separator +fileName);
            fileItem.write(uploadFile);
        }
    }
} catch (FileUploadException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
} catch (Exception e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
return articleMap;
}
}

```

BoardDAO

```

public class BoardDAO {
    Connection conn;
    private DataSource dataFactory;
    PreparedStatement pstmt;
    public BoardDAO() {
        try {
            Context ctx = new InitialContext();
            Context envctx = (Context)ctx.lookup("java:/comp/env");
            dataFactory = (DataSource) envctx.lookup("jdbc/oracle");
        } catch (NamingException e) {
            System.out.println("연결실패-----");
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }
    public void mkdirdfs(int articleNO) {
        try {
            Class.forName("org.apache.hive.jdbc.HiveDriver");
            Connection conn =
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
            Statement stmt = conn.createStatement();
            String hdfs_dir="/user/hive/warehouse/"+articleNO;

```



```

        stmt.executeQuery(sql);
        System.out.println("server에 파일추가 완료");

    } catch (ClassNotFoundException e) {
        System.out.println("class.forName실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (SQLException e) {
        System.out.println("connection실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void rmrdfs(int articleNO) {
    try {
        Class.forName("org.apache.hive.jdbc.HiveDriver");
        Connection conn
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
        Statement stmt = conn.createStatement();

        String hdfs_dir="/user/hive/warehouse/"+articleNO;

        String sql = "dfs -rmr "+hdfs_dir;
        stmt.executeQuery(sql);
        System.out.println("hadoop에서 파일삭제 완료");

    } catch (ClassNotFoundException e) {
        System.out.println("class.forName실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (SQLException e) {
        System.out.println("connection실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void rmrfiledfs(int articleNO,String filename) {
    try {
        Class.forName("org.apache.hive.jdbc.HiveDriver");
        Connection conn
DriverManager.getConnection("jdbc:hive2://192.168.30.120:10000/default", "hive", "hive");
        Statement stmt = conn.createStatement();

        String hdfs_file="/user/hive/warehouse/"+articleNO+"/"+filename;

        String sql = "dfs -rmr "+hdfs_file;
        stmt.executeQuery(sql);
        System.out.println("hadoop에서 파일삭제 완료");

    } catch (ClassNotFoundException e) {
        System.out.println("class.forName실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    } catch (SQLException e) {
        System.out.println("connection실패");
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

```



```

    }
}
public List selectAllArticles() {
    List articlesList = new ArrayList();
    try {
        conn = dataFactory.getConnection();
        String query = "select
LEVEL,articleNO,parentNO,title,content,id,writeDate" + " from t_board2"
        + " START WITH parentNO=0
CONNECT BY PRIOR articleNO=parentNO";
        //+" ORDER SIBLINGS BY
articleNO DESC";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            int level = rs.getInt("level");
            int articleNO = rs.getInt("articleNO");
            int parentNO = rs.getInt("parentNO");
            String title = rs.getString("title");
            String content = rs.getString("content");
            String id = rs.getString("id");
            Date writeDate = rs.getDate("writeDate");
            ArticleVO article = new ArticleVO();
            article.setLevel(level);
            article.setArticleNO(articleNO);
            article.setParentNO(parentNO);
            article.setTitle(title);
            article.setContent(content);
            article.setId(id);
            article.setWriteDate(writeDate);
            articlesList.add(article);
        }
        rs.close();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return articlesList;
}
public List selectmemArticles(String id) {
    List<ArticleVO> articlesList = new ArrayList<ArticleVO>();

    try {
        conn = dataFactory.getConnection();
        String query = "select distinct * from t_title2 where id=?";
        //String query = "select * from t_title2 where id=?";
        System.out.println("-----");
        System.out.println(query);
        System.out.println(id);
        System.out.println("-----");
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, id);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {

```

```

        int articleNO = rs.getInt("articleNO");
        ArticleVO articleVO = new ArticleVO();
        articleVO.setArticleNO(articleNO);
        articlesList.add(articleVO);
        System.out.println("articleNO ===== "+articleNO);
    }
    rs.close();
    pstmt.close();
    conn.close();
} catch (SQLException e) {
    // TODO Auto-generated catch block
    e.printStackTrace();
}
return articlesList;
}
/*
public List selectAllArticles() {
    List articlesList = new ArrayList();

    try {
        conn = dataFactory.getConnection();
        String query = "select
LEVEL,articleNO,parentNO,title,content,id,writeDate"+" from t_board2"
                    +" START WITH parentNO=0
CONNECT BY PRIOR articleNO=parentNO"
                    +" ORDER SIBLINGS BY
articleNO DESC";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            int level = rs.getInt("level");
            int articleNO = rs.getInt("articleNO");
            int parentNO = rs.getInt("parentNO");
            String title = rs.getString("title");
            String content = rs.getString("content");
            String id = rs.getString("id");
            Date writeDate = rs.getDate("writeDate");
            ArticleVO article = new ArticleVO();
            article.setLevel(level);
            article.setArticleNO(articleNO);
            article.setParentNO(parentNO);
            article.setTitle(title);
            article.setContent(content);
            article.setId(id);
            article.setWriteDate(writeDate);
            articlesList.add(article);
        }
        rs.close();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return articlesList;
}
}

```

```

*/
private int getNewArticleNo() {
    try {
        conn = dataFactory.getConnection();
        String query = "select max(articleNO) from t_board2";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery(query);
        if(rs.next())
            return (rs.getInt(1)+1);
        rs.close();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return 0;
}

public int insertNewArticle(ArticleVO article) {
    int articleNO = getNewArticleNo();
    try {
        conn = dataFactory.getConnection();
        int parentNO = article.getParentNO();
        String title = article.getTitle();
        String content = article.getContent();
        String id = article.getId();
        System.out.println("filename ====="+article.getFileName());
        String FileName = article.getFileName();
        String query = "INSERT INTO t_board2 (articleNO, parentNO,
title, content, filename, id)"
                                +" values (?,?,,?,?);";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.setInt(2, parentNO);
        pstmt.setString(3, title);
        pstmt.setString(4, content);
        pstmt.setString(5, FileName);
        pstmt.setString(6, id);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    return articleNO;
}

public ArticleVO selectArticle(int articleNO) {
    ArticleVO article = new ArticleVO();
    try {
        System.out.println("1
===== "+article.getFileName());
        filename

```

```

        conn = dataFactory.getConnection();
        String query = "select
articleNO,parentNO,title,content,filename,id,writeDate"
        + " from t_board2"
        + " where articleNO=?";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        ResultSet rs = pstmt.executeQuery();
        rs.next();

        int _articleNO = rs.getInt("articleNO");
        int parentNO = rs.getInt("parentNO");
        String title = rs.getString("title");
        String content = rs.getString("content");
        String FileName = rs.getString("FileName");
        String id = rs.getString("id");
        Date writeDate = rs.getDate("writeDate");

        article.setArticleNO(_articleNO);
        article.setParentNO (parentNO);
        article.setTitle(title);
        article.setContent(content);
        article.setFileName(FileName);
        article.setId(id);
        article.setWriteDate(writeDate);
        System.out.println("2filename ====="+ article.getFileName());

        rs.close();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }

    return article;
}

public void updateArticle(ArticleVO article) {
    int articleNO = article.getArticleNO();
    String title = article.getTitle();
    String content = article.getContent();
    String FileName = article.getFileName();

    try {
        conn=dataFactory.getConnection();
        String query = "update t_board2 set title=?,content=?";
        if(FileName != null && FileName.length() != 0) {
            query += ",FileName=?";
        }
        query += " where articleNO=?";

        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, title);
        pstmt.setString(2, content);

```

```

        if(fileName != null && fileName.length() != 0) {
            pstmt.setString(3, fileName);
            pstmt.setInt(4, articleNO);
        } else {
            pstmt.setInt(3, articleNO);
        }
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void deleteArticle(int articleNO) {
    try {
        conn = dataFactory.getConnection();
        String query = "DELETE FROM t_board2";
        query += " WHERE articleNO in (";
        query += " select articleNO from t_board2";
        query += " START WITH articleNO = ?";
        query += " CONNECT BY PRIOR articleNO = parentNO )";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public List<Integer> selectRemovedArticles(int articleNO){
    List<Integer> articleNOList = new ArrayList<Integer>();
    try {
        conn = dataFactory.getConnection();
        String query = "select articleNO from t_board2";
        query += " START WITH articleNO = ?";
        query += " CONNECT BY PRIOR articleNO=parentNO";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setInt(1, articleNO);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            articleNO = rs.getInt("articleNO");
            articleNOList.add(articleNO);
        }
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return articleNOList;
}
}

```

```
}
```

listArticels.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" isELIgnored="false" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<c:set var="contextPath" value="${pageContext.request.contextPath }"/>
<%
    request.setCharacterEncoding("utf-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">
<meta name="description" content="">
<meta name="author" content="">

<title>Agency - Start Bootstrap Theme</title>

<!-- Bootstrap core CSS -->
<link href="${contextPath }/page/vendor/bootstrap/css/bootstrap.min.css"
rel="stylesheet">

<!-- Custom fonts for this template -->
<link href="${contextPath }/page/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet"
type="text/css">
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Droid+Serif:400,700,400italic,700italic'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700'
rel='stylesheet' type='text/css'>

<!-- Custom styles for this template -->
<link href="${contextPath }/page/css/agency.min.css" rel="stylesheet">

<style>
    td{
        text-align:center;
    }
    td a{
        color:black;
    }
    .logout{
        width:30%;
    }
    n{
        margin:0 auto;
        font-size:1.5em;
        font-weight:bold;
    }
</style>
```

```

</head>
<body id="page-top">

  <nav class="navbar navbar-expand-lg navbar-dark fixed-top" id="mainNav">
  <div class="container">
    <a class="navbar-brand js-scroll-trigger" href="#page-top">JBHard</a>
    <button class="navbar-toggler navbar-toggler-right" type="button"
data-toggle="collapse" data-target="#navbarResponsive" aria-controls="navbarResponsive"
aria-expanded="false" aria-label="Toggle navigation">
      Menu
      <i class="fas fa-bars"></i>
    </button>
    <div class="collapse navbar-collapse" id="navbarResponsive">
    <ul class="navbar-nav text-uppercase ml-auto">
      <li class="nav-item">
        <a class="nav-link js-scroll-trigger" href="#services">프로젝트</a>
      </li>
      <c:if test="${sessionScope.id == 'admin' }">
      <li class="nav-item">
        <a class="nav-link js-scroll-trigger" href="#portfolio">회원정보</a>
        <li class="nav-item">
          <a class="nav-link js-scroll-trigger" href="${contextPath
}/board/articleForm.do">프로젝트 추가</a>
        </li>
        <li class="nav-item">
          <a class="nav-link js-scroll-trigger" href="${contextPath
}/member/memberForm.do">회원추가</a>
        </li>
      </c:if>
      <li class="nav-item">
        <a class="nav-link js-scroll-trigger" href="${contextPath }/login/logout.do">로그아웃</a>
      </li>
    </ul>
    </div>
  </div>
</nav>

  <!-- Header -->
  <header class="masthead">
  <div class="container">
    <div class="intro-text">
      <div class="intro-lead-in">Welcome To Our JBHard!</div>
      <div class="intro-heading text-uppercase">${sessionScope.id }</div>
    </div>
  </div>
</header>

  <!-- project -->
  <section class="page-section" id="services">
  <div class="container">
    <div class="row">
      <div class="col-lg-12 text-center">
        <h2 class="section-heading text-uppercase">프로젝트</h2>
        <h3 class="section-subheading text-muted">현재사용할수있는 프로젝트입니다
</h3>
      </div>
    </div>
  </div>

```

```

</div>
<div class="row">
  <table class="table table-hover">
    <c:choose>
      <c:when test="{articlesList == null }">
        <script>
          location.href="{contextPath }/login/logout.do";
        </script>
      </c:when>
      <c:when test="{articlesList !=null }">
        <c:forEach var="article" items="{articlesList }"
varStatus="articleNum">
          <c:choose>
            <c:when test='{article.level > 1 }'> <tr>
              <td>
                <a href="{contextPath }/board/viewArticle.do?articleNO={article.articleNO}">{article.title
                }</a>
              </td>
              <td> {article.writeDate }
              </td>
            </tr>
            <c:otherwise>
              <tr>
                <th colspan="3">
                  <button type="button" class="btn btn-outline-dark" style="n
onclick="location.href='{contextPath }/board/viewArticle.do?articleNO={article.articleNO}'">
                    프로젝트명 : {article.title }
                  </button>
                </th>
              </tr>
              <tr class="table-warning">
                <td>
                  <div style="text-align: right;">
                    작성자
                  </div>
                </td>
                <td>
                  <div style="text-align: right;">
                    제목
                  </div>
                </td>
                <td>
                  <div style="text-align: right;">
                    작성일
                  </div>
                </td>
              </tr>
            </c:otherwise>
          </c:choose>
        </c:forEach>
      </c:when>
    </c:choose>
  </table>
</div>
</div>
</section>
<!-- Bootstrap core JavaScript -->
<script src="<c:url value="/page/vendor/jquery/jquery.min.js" />"></script>
<script src="<c:url value="/page/vendor/bootstrap/js/bootstrap.bundle.min.js"
/>"></script>

```



```

<!-- Plugin JavaScript -->
<script src="<c:url value="/page/vendor/jquery-easing/jquery.easing.min.js" />"> </script>

<!-- Contact form JavaScript -->
<script src="<c:url value="/page/js/jqBootstrapValidation.js" />"> </script>
<script src="<c:url value="/page/js/contact_me.js" />"> </script>

<!-- Custom scripts for this template -->
<script src="<c:url value="/page/js/agency.min.js" />"> </script>
</body>
</html>

```

replyFom.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<c:set var="contextPath" value="${pageContext.request.contextPath}" />
<%
    request.setCharacterEncoding("UTF-8");
%>
<head>
<meta charset="UTF-8">
<link href="${contextPath }/page/vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">
<link href="${contextPath }/page/css/agency.css" rel="stylesheet">
<!-- Custom fonts for this template -->
<link href="${contextPath }/page/vendor/fontawesome-free/css/all.min.css" rel="stylesheet"
type="text/css">
<link href="https://fonts.googleapis.com/css?family=Montserrat:400,700" rel="stylesheet"
type="text/css">
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script' rel='stylesheet'
type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Droid+Serif:400,700,400italic,700italic'
rel='stylesheet' type='text/css'>
<link href='https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700'
rel='stylesheet' type='text/css'>
<style>
.write {
    font-size: 40px;

    text-align:center;
}
.page-section {
padding: 100px 0;
}

form {
margin: 0 auto;
width: 900px;
height: 600px;
padding: 2em;
border: 1px solid #CCC;
border-radius: 1em;
}
form div + div {

```

```

    margin-top: 1em;
  }
  label {
    display: inline-block;
    width: 100px;
    text-align:left;
  }
  input,textarea {
    font : 1em sans-serif;
    width: 400px;
    -moz-box-sizing:border-box;
    box-sizing: border-box;
    border: 1px solid #999;
  }
  textarea {
    vertical-align: top;
    height: 5em;
    resize: vertical;
  }
  #msg {
    height: 176px;
  }
  .button {
    padding-left: 90px;
  }
  button {
    margin-left: .5em;
  }
}

</style>
<script src="http://code.jquery.com/jquery-latest.min.js"></script>
<script type="text/javascript">

function backToList(obj){
    obj.action="{contextPath}/board/listArticles.do";
    obj.submit();
}

function readURL(input) {
    if (input.files && input.files[0]) {
        var reader = new FileReader();
        reader.onload = function (e) {
            $('#preview').attr('src', e.target.result);
        }
        reader.readAsDataURL(input.files[0]);
    }
}
</script>
<title>프로젝트 파일 및 글추가</title>
</head>
<body>
    <section class="page-section" id="write">
    <div class="write" >File & Write Add</div>
    <form name="frmReply" method="post" action="{contextPath}/board/addReply.do"
    enctype="multipart/form-data">
    <div>

```

```

        <label for="name">Name:</label>
        ${sessionScope.id }
    </div>
</div>
    <div>
        <label for="name">Title:</label>
        <input type="text" id="title"
name="title"/></td>
    </div>
</div>
    <div>
        <label for="content">Content:</label>
        <textarea name="content" id="msg"></textarea>
    </div>
    <div class="form-group">
        <label for="File">File Upload</label>
        <input type="file" id="File" name="FileName" onchange="readURL(this);" />
        
    </div>
    <div class="btn">
        <input type="submit" class="btn btn-primary btn-sm" value="Add"/>
        <input type="button" class="btn btn-default btn-sm" value="List"
onClick="backToList(this.form)"/>
    </div>
</form>
</section>
</body>
</html>

```

ViewArticle.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"
    isELIgnored="false" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%
    request.setCharacterEncoding("UTF-8");
%>
<c:set var="contextPath" value="${pageContext.request.contextPath}" />
<head>
    <meta charset="UTF-8">

    <title>글보기</title>
    <style>
        .img_c{
            width:100%;
        }
        .input_c{
            width:100%;
        }
        form{
            margin:0 auto;
        }
    </style>

```

```

<script src="http://code.jquery.com/jquery-latest.min.js"> </script>
<script type="text/javascript" >
    function backToList(obj){
        obj.action="{contextPath}/board/listArticles.do";
        obj.submit();
    }

    function backToList2(obj){
        obj.action="{contextPath}/member/listMembers.do";
        obj.submit();
    }

    function fn_enable(obj){
        document.getElementById("i_title").disabled=false;
        document.getElementById("i_content").disabled=false;
        document.getElementById("i_FileName").disabled=false;
        document.getElementById("tr_btn").style.display="none";
    }

    function fn_modify_article(obj){
        obj.action="{contextPath}/board/modArticle.do";
        obj.submit();
    }

    function fn_remove_article(url,articleNO){
        var form = document.createElement("form");
        form.setAttribute("method", "post");
        form.setAttribute("action", url);
        var articleNOInput = document.createElement("input");
        articleNOInput.setAttribute("type","hidden");
        articleNOInput.setAttribute("name","articleNO");
        articleNOInput.setAttribute("value", articleNO);

        form.appendChild(articleNOInput);
        document.body.appendChild(form);
        form.submit();
    }

    function fn_reply_form(url, parentNO){
        var form = document.createElement("form");
        form.setAttribute("method", "post");
        form.setAttribute("action", url);
        var parentNOInput = document.createElement("input");
        parentNOInput.setAttribute("type","hidden");
        parentNOInput.setAttribute("name","parentNO");
        parentNOInput.setAttribute("value", parentNO);

        form.appendChild(parentNOInput);
        document.body.appendChild(form);
        form.submit();
    }

    function readURL(input) {
        if (input.files && input.files[0]) {
            var reader = new FileReader();
            reader.onload = function (e) {

```

```

        $('#preview').attr('src', e.target.result);
    }
    reader.readAsDataURL(input.files[0]);
}
}
</script>
<link rel="stylesheet" href="../css/bootstrap.css">
<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.2/css/bootstrap.min.css">
</head>
<body>
<div class="container">

    <form name="frmArticle" method="post" enctype="multipart/form-data"
class="form-horizontal">
    <br>
    <br>
    <br>
    <table class="table table-striped table-bordered">
        <h1 align="center" style="font-size:60px; margin:20 auto">Project</h1>
    <tr>
        <td width="20%" align="center" style="background:#87cccb; color:#ffff">
            글번호
        </td>
        <td >
            ${article.articleNO }
            <input type="hidden" name="articleNO" value="${article.articleNO}" />
        </td>
    </tr>
    <tr>
        <td width="20%" align="center" style="background:#87cccb; color:#ffff">
            작성자
        </td>
        <td >
            ${article.id }
        </td>
    </tr>
    <tr>
        <td width="20%" align="center" style="background:#87cccb; color:#ffff">
            제목
        </td>
        <td >
            ${article.title }
        </td>
    </tr>
    <tr>
        <td width="20%" align="center" style="background:#87cccb; color:#ffff">
            내용
        </td>
        <td >
            <textarea class="input_c" rows="20" cols="60" name="content" id="i_content"
disabled >${article.content }</textarea>
        </td>
    </tr>

    <c:if test="${not empty article.fileName && article.fileName!='null' }">
    <tr>

```

```

<td width="20%" align="center" style="background:#87cccb; color:#ffff" rowspan="2">
    이미지
</td>
<td>
    <input type="hidden" name="originalFileName" value="{article.fileName }" />
     <br>
    </td>
</tr>
<tr>
<td>
    <span><b>파일명 = {article.fileName}</b></span>
    <a href="{contextPath}/download.do?FileName={article.fileName}&articleNO={article.articleNO}">파일 내려받기</a>
    </td>
</tr>
</c:if>
<tr>
<td width="20%" align="center" style="background:#87cccb; color:#ffff">
    등록일자
</td>
<td>
    {article.writeDate}
</td>
</tr>
</table>
<div align="center" style="margin:-10 auto">
    <!-- <input type=button value="수정하기" onClick="fn_enable(this.form)" -->
    <input type=button class="btn btn-info" value="삭제하기"
onClick="fn_remove_article('{contextPath}/board/removeArticle.do', {article.articleNO})">
    <c:choose>
    <c:when test="{sessionScope.id=='admin' }">
    <input type=button class="btn btn-info" value="리스트로 돌아가
기" onClick="backToList2(this.form)">
    </c:when>
    <c:otherwise>
    <input type=button class="btn btn-info" value="리스트로 돌아가
기" onClick="backToList(this.form)">
    </c:otherwise>
    </c:choose>
    <input type=button class="btn btn-info" value="파일및글추가하기"
onClick="fn_reply_form('{contextPath}/board/replyForm.do', {article.articleNO})">
</div>
</form>
</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"> </script>
<script type="text/javascript" src="js/bootstrap.js"> </script>
</body>
</html>

```

```

package log;

import java.io.IOException;
import java.io.PrintWriter;

import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;
import javax.servlet.http.HttpSession;

import mem.*;
/**
 * Servlet implementation class LoginController
 */
@WebServlet("/login/*")
public class LoginController2 extends HttpServlet {
    private static final long serialVersionUID = 1L;
    MemberService memberService;
    MemberVO memberVO;

    public LoginController2() {
        memberService = new MemberService();
    }
    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        doHandle(request,response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        doHandle(request,response);
    }

    private void doHandle(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        request.setCharacterEncoding("utf-8");
        response.setContentType("text/html;charset=utf-8");
        String action = request.getPathInfo();
        HttpSession session = request.getSession();
        String id = null;
        String pwd = null;
        PrintWriter pw = response.getWriter();

        System.out.println("session =" +session);
        if(action==null || action.equals("/loginForm.do")) {
            pw.write("<script
charsets='utf-8'>

```

```

location.href="" + request.getContextPath()+
        "/login2/loginForm.jsp";</script>");
        return;
    }else if(action.equals("/checklogin.do")) {
        id = request.getParameter("id");
        pwd = request.getParameter("pwd");
        System.out.println("id = "+id);
        System.out.println("pwd = "+pwd);

        if(session==null || session.getAttribute("id")==null) {
            memberVO = new MemberVO();
            memberVO.setId(id);
            memberVO.setPwd(pwd);
            boolean result=memberService.checklog(memberVO);

            if(result && id.equals("admin")) {
                session.setAttribute("id", id);
                pw.write("<script charset='utf-8'> alert('관리자로
로그인 하였습니다');"+ "location.href="" + request.getContextPath()
+ "/member/listMembers.do';</script> ");
                return;
            }
            if(result) {
                session.setAttribute("id", id);

                pw.write("<script charset='utf-8'> alert('"+id+"님
로그인 완료하였습니다'); location.href="" + request.getContextPath()+
"/board/listArticles.do';</script>");
                return;
            }else {
                pw.write("<script charset='utf-8'> alert('아이디 또
는 비밀번호가 틀립니다'); location.href="" + request.getContextPath()+
"/login2/loginForm.jsp';</script>");
                return;
            }
        }else if(session.getAttribute("id")!=null) {
            if(session.getAttribute("id").equals("admin")) {
                pw.write("<script charset='utf-8'> alert('관리자로
이미 로그인 하였습니다');"+ "location.href="" + request.getContextPath()
+ "/member/listMembers.do';</script> ");
                return;
            }else{
                id=(String) session.getAttribute("id");
                pw.write("<script charset='utf-8'> alert('"+id+"님
이미 로그인 하였습니다'); location.href="" + request.getContextPath()+
"/board/listArticles.do';</script>");
                return;
            }
        }else {
            System.out.println("333333333333333333333333333333");
            System.out.println("id = "+id);
            System.out.println("pwd = "+pwd);

```



```

    }
    }else if(action.equals("/logout.do")){
        session.removeAttribute("id");
        pw.write("<script charset='utf-8'> alert('로그아웃되었습니다');
location.href='"+request.getContextPath()+"/login/loginForm.do'; </script>");
        return;
    }
}
}
}

```

loginForm.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" isELIgnored="false"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>

    <c:set var="contextPath" value="${pageContext.request.contextPath }"/>
<!DOCTYPE html>
<html lang="en">
<head>
    <title>JBHard Member Login</title>
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link
        href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"
        rel="stylesheet"
        <link href='https://fonts.googleapis.com/css?family=Kaushan+Script'
rel='stylesheet' type='text/css'>
        <link rel="stylesheet" href="css/login.css">
        <style type="text/css">
    body {
        background: #ffff;
        font-family: arial;
    }
    .login-form h1 {
        font-size: 50px;
        text-align: center;
        color: #fed136;
        margin-bottom: 30px;
        font-weight: normal;
    }
    .login-form .social-icon {
        width: 100%;
        font-size: 20px;
        padding-top: 20px;
        color: #fff;
        text-align: center;
        float: left;
    }
    .login-form {
        background: #fff;
        width: 450px;
        border-radius: 6px;
        margin: 0 auto;
        display: table;
        padding: 15px 30px 30px;

```

```

    box-sizing: border-box;
}
.form-group {
    float: left;
    width: 100%;
    margin: 0 0 15px;
    position: relative;
}
.login-form input {
    width: 100%;
    padding: 5px;
    height: 56px;
    border-radius: 74px;
    border: 1px solid #fed136;
    box-sizing: border-box;
    font-size: 15px;
    padding-left: 75px;
}
.login-form .form-group .input-icon {
    font-size: 15px;
    display: -webkit-box;
    display: -webkit-flex;
    display: -moz-box;
    display: -ms-flexbox;
    display: flex;
    align-items: center;
    position: absolute;
    border-radius: 25px;
    bottom: 0;
    height: 100%;
    padding-left: 35px;
    color: #666;
}
.login-form .login-btn {
    background: #fed136;
    padding: 11px 50px;
    border-color: #fff545;
    color: #fff;
    text-align: center;
    margin: 0 auto;
    font-size: 20px;
    border: 1px solid #fed136;
    border-radius: 44px;
    width: 100%;
    height: 56px;
    cursor: pointer;
}
.login-form .reset-psw {
    float: left;
    width: 100%;
    text-decoration: none;
    color: #756f05;
    font-size: 14px;
    text-align: center;
    margin-top: 11px;
}
.login-form .social-icon button {

```

```

font-size: 20px;
color: white;
height: 50px;
width: 50px;
background: #ffd400;
border-radius: 60%;
margin: 0px 10px;
border: none;
cursor: pointer;
}
.login-form button:hover{
  opacity: 0.9;
}
.login-form .seperator {
  float: left;
  width: 100%;
  border-top: 1px solid #fed136;
  text-align: center;
  margin: 50px 0 0;
}
.login-form .seperator b {
  width: 40px;
  height: 40px;
  font-size: 17px;
  text-align: center;
  line-height: 37px;
  background: #fff;
  display: inline-block;
  border: 1px solid #fed136;
  border-radius: 55%;
  position: relative;
  top: -22px;
  z-index: 1;
}
.login-form p {
  color: #756f05;
  float: left;
  width: 100%;
  text-align: center;
  font-size: 16px;
  margin: 0 0 10px;
}
}
@media screen and (max-width:767px) {
.login-form {
  width: 90%;
  padding: 15px 15px 30px;
}
}

.login-form {
  width: 600px;
  margin: 120px auto;
  text-align: center;
}
.login-form .login {
  font-family: 'Kaushan Script', -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto,
'Helvetica Neue', Arial, sans-serif, 'Apple Color Emoji', 'Segoe UI Emoji', 'Segoe UI Symbol',
'Noto Color Emoji';

```

```

    }
    }
    </style>
</head>
<body>
    <div class="login-form">
        <form method="post" action="{contextPath }/login/checklogin.do"
encType="utf-8">
            <h1 class="login">JB-Hard</h1>
            <div class="form-group">
                <input type="text" name="id" placeholder="아이디를 입
력해주세요">
                <span class="input-icon"><i class="fa fa-user"></i></span>
            </div>
            <div class="form-group">
                <input type="password" name="pwd" placeholder="비밀
번호를 입력해주세요"/>
                <span class="input-icon"><i class="fa fa-lock"></i></span>
            </div>
            <button type="submit" class="login-btn">Login</button>

        </form>
    </div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
<script type="text/javascript">
    // JS not require for this snippet
</script>
</body>
</html>

```

listMembers.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8" import="java.util.*,mem.*" isELIgnored="false" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<c:set var="contextPath" value="{pageContext.request.contextPath }"/>
<%
    request.setCharacterEncoding("utf-8");
%>

<!DOCTYPE html>
<html>
<head>
<c:choose>
    <c:when test="{msg=="addMember" }">
        <script>
            window.onload=function(){
                alert("회원을 등록했습니다");
            }
        </script>
    </c:when>
    <c:when test="{msg=="modified" }">
        <script>
            window.onload=function(){

```

```

                                alert("회원 정보를 수정했습니다");
                            }
                        </script>
                    </c:when>
                    <c:when test='${msg=="deleted" }'>
                        <script>
                            window.onload=function(){
                                alert("회원 정보를 삭제했습니다");
                            }
                        </script>
                    </c:when>
                </c:choose>
                <meta charset="UTF-8">
                <title>Insert title here</title>
                <style>
                    td{
                        text-align:center;
                    }
                    .n{
                        width:100%;
                        font-size:2em;
                        font-weight:bold;
                    }
                </style>

                <!-- Bootstrap core CSS -->
                <link rel="stylesheet" href="{contextPath
}/page/vendor/bootstrap/css/bootstrap.min.css">

                <!-- Custom fonts for this template -->
                <link rel="stylesheet" type="text/css" href="{contextPath
}/page/vendor/fontawesome-free/css/all.min.css">
                <link rel="stylesheet" type="text/css"
href="https://fonts.googleapis.com/css?family=Montserrat:400,700">
                <link rel="stylesheet" type="text/css"
href='https://fonts.googleapis.com/css?family=Kaushan+Script' >
                <link rel="stylesheet" type="text/css"
href='https://fonts.googleapis.com/css?family=Droid+Serif:400,700,400italic,700italic' >
                <link rel="stylesheet" type="text/css"
href='https://fonts.googleapis.com/css?family=Roboto+Slab:400,100,300,700'>

                <!-- Custom styles for this template -->
                <link rel="stylesheet" href="{contextPath }/page/css/agency.min.css">
            </head>
            <body>
            <c:import url="/board/listArticles.do"> </c:import>

            <!-- information -->
            <section class="bg-light page-section" id="portfolio">
                <div class="container">
                    <div class="row">
                        <div class="col-lg-12 text-center">
                            <h2 class="section-heading text-uppercase">회원정보</h2>
                            <h3 class="section-subheading text-muted">회원정보를 수정할수있습니다</h3>
                        </div>
                    </div>
                </div>
            <div class="row">

```

```

<table class="table table-hover">
  <tr class="table-warning">
    <td>아이디</td>
    <td>비밀번호</td>
    <td>이름</td>
    <td>이메일</td>
    <td>가입일</td>
    <td>수정</td>
    <td>삭제</td>
  </tr>

  <c:choose>
    <c:when test="{membersList==null }">
      <tr>
        <td colspan=5>
          <b>등록된 회원이 없습니다.</b>
        </td>
      </tr>
    </c:when>
    <c:when test="{membersList!=null }">
      <c:forEach var="data" items="{membersList }">
        <tr align="center">
          <td>${data.id }</td>
          <td>${data.pwd }</td>
          <td>${data.name }</td>
          <td>${data.email }</td>
          <td>${data.joinDate }</td>
          <td><a href="{contextPath
}/member/modMemberForm.do?id=${data.id}">수정</a></td>
          <td><a href="{contextPath
}/member/delMember.do?id=${data.id}">삭제</a></td>
        </tr>
      </c:forEach>
    </c:when>
  </c:choose>
</table>
</div>
</section>

<!-- Bootstrap core JavaScript -->
<script src="{c:url value="/page/vendor/jquery/jquery.min.js" />"></script>
<script src="{c:url value="/page/vendor/bootstrap/js/bootstrap.bundle.min.js"
/>"></script>

<!-- Plugin JavaScript -->
<script src="{c:url value="/page/vendor/jquery-easing/jquery.easing.min.js" />"></script>

<!-- Contact form JavaScript -->
<script src="{c:url value="/page/js/jqBootstrapValidation.js" />"></script>
<script src="{c:url value="/page/js/contact_me.js" />"></script>

<!-- Custom scripts for this template -->
<script src="{c:url value="/page/js/agency.min.js" />"></script>
</body>
</html>

```

MemberController

```
package mem;

import java.io.IOException;
import java.util.List;

import javax.servlet.RequestDispatcher;
import javax.servlet.ServletConfig;
import javax.servlet.ServletException;
import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

/**
 * Servlet implementation class MemberController2
 */
@WebServlet("/member/*")
public class MemberController3 extends HttpServlet {
    private static final long serialVersionUID = 1L;
    MemberDAO memberDAO;

    /**
     * @see Servlet#init(ServletConfig)
     */
    public void init(ServletConfig config) throws ServletException {
        memberDAO = new MemberDAO();
    }

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        doHandle(request,response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
    response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
    throws ServletException, IOException {
        doHandle(request,response);
    }

    private void doHandle(HttpServletRequest request,HttpServletResponse response)
    throws ServletException, IOException{
        String nextPage = null;
        request.setCharacterEncoding("utf-8");
        response.setContentType("text/html;charset=utf-8");
        String action = request.getPathInfo();
        System.out.println("action : "+action);
        if(action == null || action.equals("/listMembers.do")){
            List<MemberVO> membersList = memberDAO.listMembers();

```

```

        request.setAttribute("membersList", membersList);
        nextPage = "/member2/listMembers.jsp";
    }else if(action.equals("/addMember.do")) {
        String id = request.getParameter("id");
        String pwd = request.getParameter("pwd");
        String name = request.getParameter("name");
        String email = request.getParameter("email");
        MemberVO memberVO = new MemberVO(id,pwd,name,email);
        memberDAO.addMember(memberVO);
        nextPage = "/member/listMembers.do";

    }else if(action.equals("/memberForm.do")) {
        nextPage = "/member2/memberForm.jsp";
    }else if(action.equals("/modMemberForm.do")) {
        String id = request.getParameter("id");
        System.out.println("id
"+id+"-----");
        MemberVO memInfo = memberDAO.findMember(id);
        request.setAttribute("memInfo", memInfo);
        nextPage="/member2/modMemberForm.jsp";
    }else if(action.equals("/modMember.do")){
        String id = request.getParameter("id");
        String pwd = request.getParameter("pwd");
        String name = request.getParameter("name");
        String email = request.getParameter("email");
        System.out.println("name(controller
"+name+"-----");
        System.out.println("id(controller = "+id+"-----");
        System.out.println("email(controller
"+email+"-----");
        MemberVO memberVO = new MemberVO(id,pwd,name,email);
        memberDAO.modMember(memberVO);
        request.setAttribute("msg", "modified");
        nextPage = "/member/listMembers.do";
    }else if(action.equals("/delMember.do")){
        String id = request.getParameter("id");
        memberDAO.delMember(id);
        request.setAttribute("msg", "deleted");
        nextPage = "/member/listMembers.do";
    }else {
        List membersList = memberDAO.listMembers();
        request.setAttribute("membersList", membersList);
        nextPage = "/member2/listMembers.jsp";
    }
    RequestDispatcher dispatch = request.getRequestDispatcher(nextPage);
    dispatch.forward(request, response);
}
}

```

MemberDAO

```

package mem;

import java.sql.Connection;
import java.sql.Date;
import java.sql.PreparedStatement;

```



```

import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

import javax.naming.Context;
import javax.naming.InitialContext;
import javax.naming.NamingException;
import javax.sql.DataSource;

public class MemberDAO {
    private DataSource dataFactory;
    private Connection conn;
    private PreparedStatement pstmt;

    public MemberDAO() {
        try {
            Context ctx = new InitialContext();
            Context envctx = (Context) ctx.lookup("java:/comp/env");
            dataFactory = (DataSource) envctx.lookup("jdbc/oracle");
        } catch (NamingException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
    }

    public List listMembers() {
        List<MemberVO> membersList = new ArrayList();

        try {
            conn = dataFactory.getConnection();
            String query = "select * from t_member2 order by joinDate
desc";

            System.out.println(query);
            pstmt=conn.prepareStatement(query);
            ResultSet rs = pstmt.executeQuery();
            while(rs.next()) {
                MemberVO vo = new MemberVO();
                String id = rs.getString("id");
                String pwd = rs.getString("pwd");
                String name = rs.getString("name");
                String email = rs.getString("email");
                Date joinDate = rs.getDate("joinDate");
                MemberVO memberVO = new
MemberVO(id,pwd,name,email,joinDate);
                membersList.add(memberVO);
            }
            rs.close();
            pstmt.close();
            conn.close();
        } catch (SQLException e) {
            // TODO Auto-generated catch block
            e.printStackTrace();
        }
        return membersList;
    }
}

```

```

public List selectMembersid() {
    List<MemberVO> membersList = new ArrayList();

    try {
        conn = dataFactory.getConnection();
        String query = "select id from t_member2 order by joinDate
desc";

        System.out.println(query);
        pstmt=conn.prepareStatement(query);
        ResultSet rs = pstmt.executeQuery();
        while(rs.next()) {
            MemberVO vo = new MemberVO();
            String id = rs.getString("id");
            vo.setId(id);
            membersList.add(vo);
        }
        rs.close();
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
    return membersList;
}

public void addMember(MemberVO m) {
    try {
        conn = dataFactory.getConnection();

        String id=m.getId();
        String pwd=m.getPwd();
        String name=m.getName();
        String email=m.getEmail();

        String query = "insert into t_member2(id,pwd,name,email)
values(?,?,?,?)";

        System.out.println(query);
        pstmt=conn.prepareStatement(query);
        pstmt.setString(1, id);
        pstmt.setString(2, pwd);
        pstmt.setString(3, name);
        pstmt.setString(4, email);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();

    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public void addtitleMem(int articleNO,List<MemberVO> membersList) {
    try {

```

```

        conn = dataFactory.getConnection();
        String id=null;
        String query = "insert into t_title2(articleNO,id) values(?,?)";
        System.out.println(query);
        pstmt=conn.prepareStatement(query);

        for(MemberVO vo : membersList) {
            id = vo.getId();
            pstmt.setInt(1, articleNO);
            pstmt.setString(2, id);
            pstmt.executeUpdate();
        }
        pstmt.close();
        conn.close();
    } catch (SQLException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
    }
}

public MemberVO findMember(String _id) {
    MemberVO meminfo = null;
    try {
        conn = dataFactory.getConnection();
        String query = "select * from t_member2 where id=?";
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, _id);
        System.out.println(query);
        ResultSet rs = pstmt.executeQuery();
        rs.next();
        String id = rs.getString("id");
        String pwd = rs.getString("pwd");
        String name = rs.getString("name");
        String email = rs.getString("email");
        Date joinDate = rs.getDate("joinDate");
        meminfo = new MemberVO(id,pwd,name,email,joinDate);
        pstmt.close();
        conn.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
    return meminfo;
}

public boolean isExisted(MemberVO memberVO) {
    boolean result = false;
    String id = memberVO.getId();
    String pwd = memberVO.getPwd();

    try {
        conn = dataFactory.getConnection();
        String query = "select decode(count(*),1,'true','false') as result
from t_member2";
        query += " where id=? and pwd=?";
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, id);
        pstmt.setString(2, pwd);
        ResultSet rs = pstmt.executeQuery();

```

```

        rs.next();
        result = Boolean.parseBoolean(rs.getString("result"));
        System.out.println("result="+result);
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
    return result;
}
}
public void modMember(MemberVO memberVO) {
    String id = memberVO.getId();
    String pwd = memberVO.getPwd();
    String name = memberVO.getName();
    String email = memberVO.getEmail();
    System.out.println("id = "+id+"---+++++-----");
    System.out.println("name = "+name+"---+++++-----");
    System.out.println("email = "+email+"---+++++-----");
    try {
        conn = dataFactory.getConnection();
        String query = "update t_member2 set pwd=?,name=?,email=?
where id=?";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, pwd);
        pstmt.setString(2, name);
        pstmt.setString(3, email);
        pstmt.setString(4, id);
        pstmt.executeUpdate();
        pstmt.close();
        conn.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
public void delMember(String id) {
    try {
        conn = dataFactory.getConnection();
        String query = "delete from t_member2 where id=?";
        System.out.println(query);
        pstmt = conn.prepareStatement(query);
        pstmt.setString(1, id);
        pstmt.executeUpdate();
    } catch (Exception e) {
        // TODO: handle exception
    }
}
}
}
}

```

memberForm.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8" isELIgnored="false" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<c:set var="contextPath" value="${pageContext.request.contextPath}"/>

```

```

<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1">
<link rel="stylesheet"
href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script'
rel='stylesheet' type='text/css'>
<link rel="stylesheet" href="css/login.css">
<style type="text/css">
body {
background: #ffff;
font-family: arial;
}
.login-form h1 {
font-size: 50px;
text-align: center;
color: #808080;
margin-bottom: 30px;
font-weight: normal;
}
.login-form .social-icon {
width: 100%;
font-size: 20px;
padding-top: 20px;
color: #fff;
text-align: center;
float: left;
}
.login-form {
background: #fff;
width: 450px;
border-radius: 6px;
margin: 0 auto;
display: table;
padding: 15px 30px 30px;
box-sizing: border-box;
}
.form-group {
float: left;
width: 100%;
margin: 0 0 15px;
position: relative;
}
.login-form input {
width: 100%;
padding: 5px;
height: 65px;
border-radius: 74px;
border: 1px solid #808080;
box-sizing: border-box;
font-size: 15px;
padding-left: 75px;
}
.login-form .form-group .input-icon {
font-size: 15px;

```

```

display: -webkit-box;
display: -webkit-flex;
display: -moz-box;
display: -ms-flexbox;
display: flex;
align-items: center;
position: absolute;
border-radius: 25px;
bottom: 0;
height: 100%;
padding-left: 35px;
color: #666;
}
.login-form .login-btn {
background: #87cccb;
padding: 11px 50px;
border-color: #87cccb;
color: #fff;
text-align: center;
margin: 0 auto;
font-size: 20px;
border: 1px solid #87cccb;
border-radius: 44px;
width: 30%;
height: 56px;
cursor: pointer;
}
.login-bttn {
background: #ffffff;
padding: 11px 50px;
border-color: #808080;
color: #000000;
text-align: center;
margin: 0 auto;
font-size: 20px;
border: 1px solid #808080;
border-radius: 44px;
width: 30%;
height: 56px;
cursor: pointer;
}
.login-form .reset-psw {
float: left;
width: 100%;
text-decoration: none;
color: #756f05;
font-size: 14px;
text-align: center;
margin-top: 11px;
}
.login-form .social-icon button {
font-size: 20px;
color: white;
height: 50px;
width: 50px;
background: #ffd400;
border-radius: 60%;

```

```

margin: 0px 10px;
border: none;
cursor: pointer;
}
.login-form button:hover{
opacity: 0.9;
}
.login-form .seperator {
float: left;
width: 100%;
border-top: 1px solid #fed136;
text-align: center;
margin: 50px 0 0;
}
.login-form .seperator b {
width: 40px;
height: 40px;
font-size: 17px;
text-align: center;
line-height: 37px;
background: #fff;
display: inline-block;
border: 1px solid #fed136;
border-radius: 55%;
position: relative;
top: -22px;
z-index: 1;
}
.login-form p {
color: #756f05;
float: left;
width: 100%;
text-align: center;
font-size: 16px;
margin: 0 0 10px;
}
}
@media screen and (max-width:767px) {
.login-form {
width: 90%;
padding: 15px 15px 30px;
}
}

.login-form {
width: 900px;
margin: 120px auto;
text-align: center;
}
.login-form .login {
font-family: 'Kaushan Script', -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto,
'Helvetica Neue', Arial, sans-serif, 'Apple Color Emoji', 'Segoe UI Emoji', 'Segoe UI Symbol',
'Noto Color Emoji';
}
}
</style>
</head>
<body>
<div class="login-form">

```

```

<form method="post" action="{contextPath }/member/addMember.do">
    <h1 class="login">Members</h1>
    <div class="form-group">
        <input type="text" name="id" placeholder="ID">
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>
    <div class="form-group">
        <input type="password" name="pwd" placeholder="Password"/>
        <span class="input-icon"><i class="fa fa-lock"></i></span>
    </div>
    <div class="form-group">
        <input type="text" name="name" placeholder="Name">
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>
    <div class="form-group">
        <input type="text" name="email" placeholder="test@test.com">
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>
    <button type="submit" class="login-btn">Add</button>
    <button type="reset" class="login-btnn">Cancel</button>
</form>
</div>
</body>
</html>

```

MemberService

```

package mem;

public class MemberService {
    MemberDAO memDAO;

    public MemberService() {
        memDAO = new MemberDAO();
    }

    public boolean checklog(MemberVO memberVO) {
        boolean result = memDAO.isExisted(memberVO);
        return result;
    }
}

```

MemberVO

```

package mem;

import java.sql.Date;

public class MemberVO {
    private String id;
    public String getId() {
        return id;
    }
    public void setId(String id) {
        this.id = id;
    }
}

```



```

public String getPwd() {
    return pwd;
}
public void setPwd(String pwd) {
    this.pwd = pwd;
}
public String getName() {
    return name;
}
public void setName(String name) {
    this.name = name;
}
public String getEmail() {
    return email;
}
public void setEmail(String email) {
    this.email = email;
}

public Date getJoinDate() {
    return joinDate;
}

public void setJoinDate(Date joinDate) {
    this.joinDate = joinDate;
}

private String pwd;
private String name;
private String email;
private Date joinDate;

public MemberVO() {
    System.out.println("MemberVO 생성자 호출");
}

public MemberVO(String id,String pwd,String name,String email) {
    super();
    this.id=id;
    this.pwd=pwd;
    this.name=name;
    this.email=email;
}

public MemberVO(String id,String pwd,String name,String email,Date joinDate) {
    super();
    this.id=id;
    this.pwd=pwd;
    this.name=name;
    this.email=email;
    this.joinDate=joinDate;
}
}

```

modMemberForm.jsp

```

<%@ page language="java" contentType="text/html; charset=UTF-8"
pageEncoding="UTF-8" isELIgnored="false" %>
<%@ taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<c:set var="contextPath" value="{pageContext.request.contextPath }"/>
<%
    request.setCharacterEncoding("utf-8");
%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
<meta name="viewport" content="width=device-width, initial-scale=1">
<link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css"
rel="stylesheet"
<link href='https://fonts.googleapis.com/css?family=Kaushan+Script'
rel='stylesheet' type='text/css'>
<link rel="stylesheet" href="css/login.css">
<style type="text/css">
    body {
        background: #ffff;
        font-family: arial;
    }
    .login-form h1 {
        font-size: 50px;
        text-align: center;
        color: #808080;
        margin-bottom: 30px;
        font-weight: normal;
    }
    .login-form .social-icon {
        width: 100%;
        font-size: 20px;
        padding-top: 20px;
        color: #fff;
        text-align: center;
        float: left;
    }
    .login-form {
        background: #fff;
        width: 450px;
        border-radius: 6px;
        margin: 0 auto;
        display: table;
        padding: 15px 30px 30px;
        box-sizing: border-box;
    }
    .form-group {
        float: left;
        width: 100%;
        margin: 0 0 15px;
        position: relative;
    }
    .login-form input {
        width: 100%;
        padding: 5px;

```

```

height: 65px;
border-radius: 74px;
border: 1px solid #808080;
box-sizing: border-box;
font-size: 15px;
padding-left: 75px;
}
.login-form .form-group .input-icon {
font-size: 15px;
display: -webkit-box;
display: -webkit-flex;
display: -moz-box;
display: -ms-flexbox;
display: flex;
align-items: center;
position: absolute;
border-radius: 25px;
bottom: 0;
height: 100%;
padding-left: 35px;
color: #666;
}
.login-form .login-btn {
background: #87cccb;
padding: 11px 50px;
border-color: #87cccb;
color: #fff;
text-align: center;
margin: 0 auto;
font-size: 20px;
border: 1px solid #87cccb;
border-radius: 44px;
width: 30%;
height: 56px;
cursor: pointer;
}
.login-bttn {
background: #ffffff;
padding: 11px 50px;
border-color: #808080;
color: #000000;
text-align: center;
margin: 0 auto;
font-size: 20px;
border: 1px solid #808080;
border-radius: 44px;
width: 30%;
height: 56px;
cursor: pointer;
}
.login-form .reset-psw {
float: left;
width: 100%;
text-decoration: none;
color: #756f05;
font-size: 14px;
text-align: center;

```

```

margin-top: 11px;
}
.login-form .social-icon button {
font-size: 20px;
color: white;
height: 50px;
width: 50px;
background: #ffd400;
border-radius: 60%;
margin: 0px 10px;
border: none;
cursor: pointer;
}
.login-form button:hover{
opacity: 0.9;
}
.login-form .seperator {
float: left;
width: 100%;
border-top: 1px solid #fed136;
text-align: center;
margin: 50px 0 0;
}
.login-form .seperator b {
width: 40px;
height: 40px;
font-size: 17px;
text-align: center;
line-height: 37px;
background: #fff;
display: inline-block;
border: 1px solid #fed136;
border-radius: 55%;
position: relative;
top: -22px;
z-index: 1;
}
.login-form p {
color: #756f05;
float: left;
width: 100%;
text-align: center;
font-size: 16px;
margin: 0 0 10px;
}
@media screen and (max-width:767px) {
.login-form {
width: 90%;
padding: 15px 15px 30px;
}
}

.login-form {
width: 900px;
margin: 120px auto;
text-align: center;
}
.login-form .login {

```

```

        font-family: 'Kaushan Script', -apple-system, BlinkMacSystemFont, 'Segoe UI', Roboto,
'Helvetica Neue', Arial, sans-serif, 'Apple Color Emoji', 'Segoe UI Emoji', 'Segoe UI Symbol',
'Noto Color Emoji';
    }
}
</style>
</head>
<body>
<div class="login-form">

<form method="post" action="{contextPath }/member/modMember.do?id={memInfo.id}">
    <h1 class="login">Member</h1>
    <div class="form-group">
        <input type="text" name="id" value="{memInfo.id }" disabled>
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>

    <div class="form-group">
        <input type="password" name="pwd" placeholder="Password"/>
        <span class="input-icon"><i class="fa fa-lock"></i></span>
    </div>
    <div class="form-group">
        <input type="text" name="name" placeholder="Name">
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>
    <div class="form-group">
        <input type="text" name="email" placeholder="test@test.com">
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>
    <div class="form-group">
        <input type="text" name="joinDate" value="{memInfo.joinDate }"
disabled>
        <span class="input-icon"><i class="fa fa-user"></i></span>
    </div>

    <button type="submit" class="login-btn">Edit</button>
    <button type="reset" class="login-bttn">Cancel</button>

    </form>
</div>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>
<script type="text/javascript" src="js/bootstrap.js"></script>
</body>
</html>

```

FileDownloadController

```

package common;

import java.io.File;
import java.io.FileInputStream;
import java.io.IOException;
import java.io.OutputStream;

import javax.servlet.ServletException;

```

```

import javax.servlet.annotation.WebServlet;
import javax.servlet.http.HttpServlet;
import javax.servlet.http.HttpServletRequest;
import javax.servlet.http.HttpServletResponse;

import org.apache.commons.io.FileUtils;

import brd.BoardService;

/**
 * Servlet implementation class FileDownloadController
 */
@WebServlet("/download.do")
public class FileDownloadController extends HttpServlet {
    private static final long serialVersionUID = 1L;
    private static String ARTICLE_IMAGE_REPO = File.separator + "dir";

    /**
     * @see HttpServlet#doGet(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doGet(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        doHandle(request,response);
    }

    /**
     * @see HttpServlet#doPost(HttpServletRequest request, HttpServletResponse
response)
     */
    protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
        doHandle(request,response);
    }

    private void doHandle(HttpServletRequest request,HttpServletResponse response)
throws ServletException, IOException{
        request.setCharacterEncoding("UTF-8");
        response.setContentType("text/html;charset=utf-8");
        if(request.getParameter("FileName")==null) {
            System.out.println("FileName =
"+request.getParameter("FileName"));
            return;
        }
        String fileName = request.getParameter("FileName");
        String articleNO = request.getParameter("articleNO");
        System.out.println("3 FileName="+fileName);
        System.out.println("FILEDOWNLOAD articleNO = "+articleNO);
        BoardService boardservice = new BoardService();

        OutputStream out = response.getOutputStream();
        String path = ARTICLE_IMAGE_REPO + File.separator + articleNO +
File.separator + fileName;
        File File_n = new File(path);
        if(!File_n.exists()) {
            boardservice.intoserver(articleNO, fileName);
        }
    }
}

```

